On the Deployment of VoIP in Ethernet Networks: Methodology and Case Study

Khaled Salah[**]
Department of Information and Computer Science
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia
Email: salah@kfupm.edu.sa

Abstract

Deploying IP telephony or voice over IP (VoIP) is a major and challenging task for data network researchers and designers. This paper outlines guidelines and a step-by-step methodology on how VoIP can be deployed successfully. The methodology can be used to assess the support and readiness of an existing network. Prior to the purchase and deployment of VoIP equipment, the methodology predicts the number of VoIP calls that can be sustained by an existing network while satisfying QoS requirements of all network services and leaving adequate capacity for future growth. As a case study, we apply the methodology steps on a typical network of a small enterprise. We utilize both analysis and simulation to investigate throughput and delay bounds. Our analysis is based on queueing theory, and OPNET is used for simulation. Results obtained from analysis and simulation are in line and give a close match. In addition, the paper discusses many design and engineering issues. These issues include characteristics of VoIP traffic and QoS requirements, VoIP flow and call distribution, defining future growth capacity, and measurement and impact of background traffic.

Keywords: Network Design, Network Management, VoIP, Performance Evaluation, Analysis, Simulation, OPNET

1 Introduction

These days a massive deployment of VoIP is taking place over data networks. Most of these networks are Ethernet-based and running IP protocol. Many network managers are finding it very attractive and cost effective to merge and unify voice and data networks into one. It is easier to run, manage, and maintain. However, one has to keep in mind that IP networks are best-effort networks that were designed for non-real time applications. On the other hand, VoIP requires timely packet delivery with low latency, jitter, packet loss, and sufficient bandwidth. To achieve this goal, an efficient deployment of VoIP must ensure these real-time traffic requirements can be guaranteed over new or existing IP networks.

[**] Corresponding Author: Prof. K. Salah, PO Box 5066, ICS Department, KFUPM, Dhahran 31261, Saudi Arabia

When deploying a new network service such as VoIP over existing network, many network architects, managers, planners, designers, and engineers are faced with common strategic, and sometimes challenging, questions. What are the QoS requirements for VoIP? How will the new VoIP load impact the QoS for currently running network services and applications? Will my existing network support VoIP and satisfy the standardized QoS requirements? If so, how many VoIP calls can the network support before upgrading prematurely any part of the existing network hardware?

These challenging questions have led to the development of some commercial tools for testing the performance of multimedia applications in data networks. A list of the available commercial tools that support VoIP is listed in [1,2]. For the most part, these tools use two common approaches in assessing the deployment of VoIP into the existing network. One approach is based on first performing network measurements and then predicting the network readiness for supporting VoIP. The prediction of the network readiness is based on assessing the health of network elements. The second approach is based on injecting real VoIP traffic into existing network and measuring the resulting delay, jitter, and loss.

Other than the cost associated with the commercial tools, none of the commercial tools offer a comprehensive approach for successful VoIP deployment. In particular, none gives any prediction for the total number of calls that can be supported by the network taking into account important design and engineering factors. These factors include VoIP flow and call distribution, future growth capacity, performance thresholds, impact of VoIP on existing network services and applications, and impact background traffic on VoIP. This paper attempts to address those important factors and layout a comprehensive methodology for a successful deployment of any multimedia application such as VoIP and videoconferencing. However, the paper focuses on VoIP as the new service of interest to be deployed. The paper also contains many useful engineering and design guidelines, and discusses many practical issues pertaining to the deployment of VoIP. These issues include characteristics of VoIP traffic and QoS requirements, VoIP flow and call distribution, defining future growth capacity, and measurement and impact of background traffic. As a case study, we illustrate how our approach and guidelines can be applied to a typical network of a small enterprise.

The rest of the paper is organized as follows. Section 2 presents a typical network topology of a small enterprise to be used as a case study for deploying VoIP. Section 3 outlines practical eight-step methodology to deploy successfully VoIP in data networks. Each step is described in considerable detail. Section 4 describes important design and engineering decisions to be made based on the analytic and simulation studies. Section 5 concludes the study and identifies future work.

2 Existing Network

Figure 1 illustrates a typical network topology for a small enterprise residing in a high-rise building. The network shown is realistic and used as a case study only; however, our work presented in this paper can be adopted *easily* for larger and general networks by following the same principles, guidelines, and concepts laid out in this paper. The network is Ethernet-based and has two Layer-2 Ethernet switches connected by a router. The router is Cisco 2621, and the switches are 3Com Superstack 3300. Switch 1 connects Floor 1 and Floor 2

and two servers; while Switch 2 connects Floor 3 and four servers. Each floor LAN is basically a shared Ethernet connecting employee PCs with workgroup and printer servers. The network makes use of VLANs in order to isolate broadcast and multicast traffic. A total of five LANs exist. All VLANs are port based. Switch 1 is configured such that it has three VLANs. VLAN1 includes the database and file servers. VLAN2 includes Floor 1. VLAN3 includes Floor2. On the other hand, Switch 2 is configured to have two VLANs. VLAN4 includes the servers for E-mail, HTTP, Web & cache proxy, and firewall. VLAN5 includes Floor 3. All the links are switched Ethernet 100Mbps full duplex except for the links for Floor 1, Floor 2, and Floor 3 which are shared Ethernet 100Mbps half duplex.

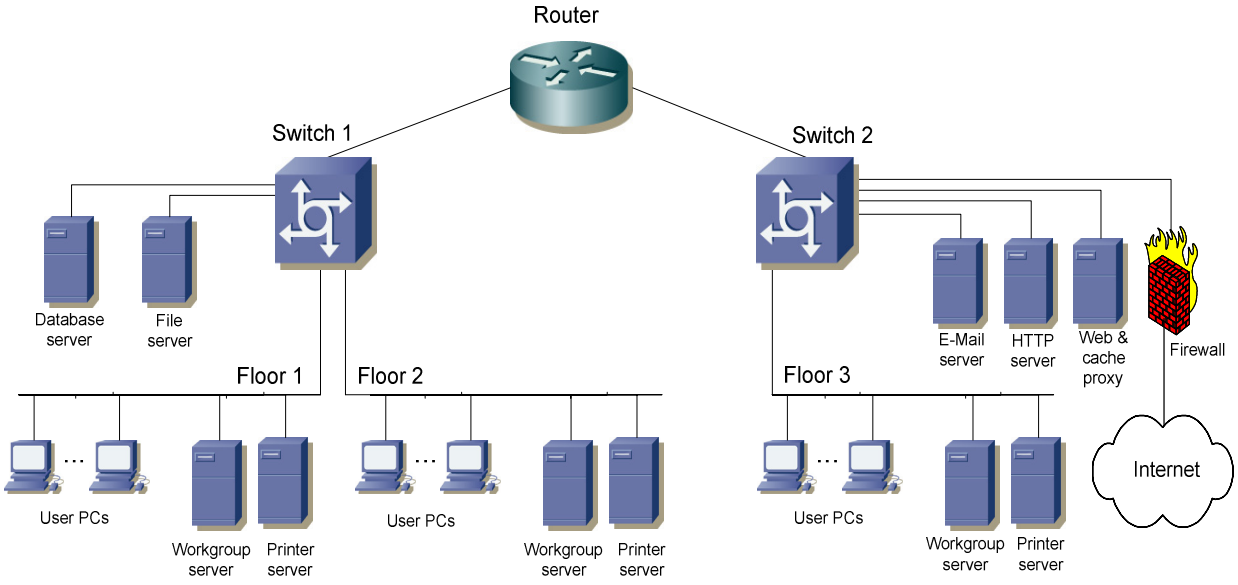


Figure 1. Logical diagram of a small enterprise

3 Step-by-Step Methodology

Figure 2 shows a flowchart of a methodology of eight steps for a successful VoIP deployment. The first four steps are independent and can be performed in parallel. Before embarking on the analysis and simulation study, in Step 6 and Step 7, Step 5 must be carried out which requires any early and necessary redimensioning or modifications to the existing network. As shown, both Step 6 and Step 7 can be done in parallel. The final step is pilot deployment.

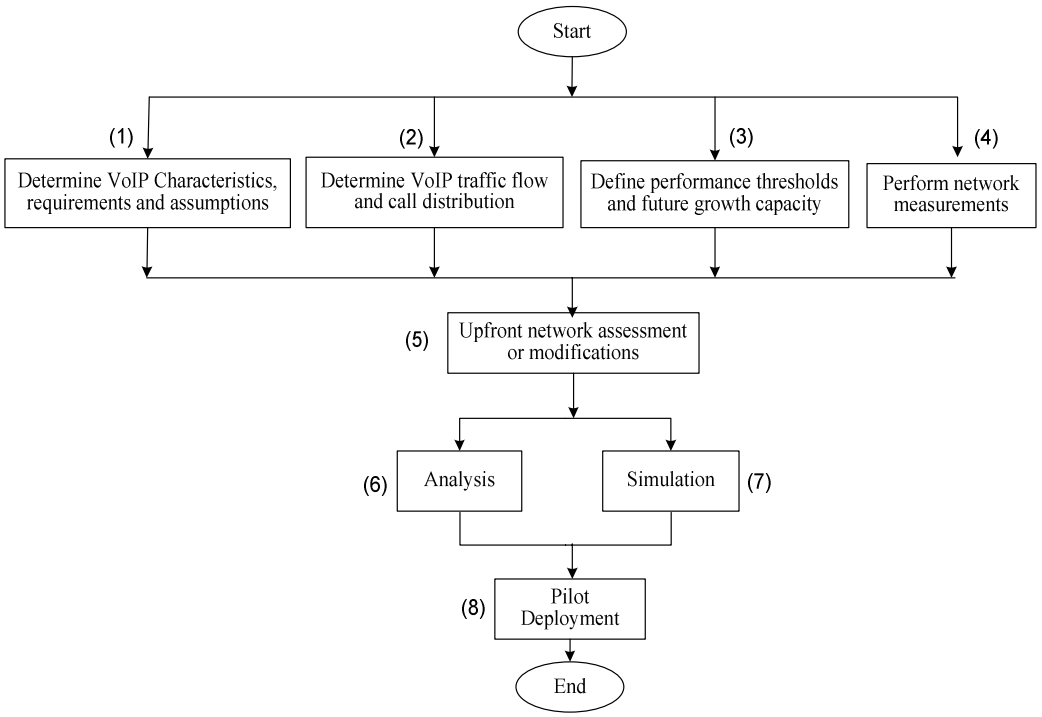


Figure 2. Flowchart illustrating methodology steps

3.1 VoIP Traffic Characteristics, Requirements, and Assumptions

For introducing a new network service such as VoIP, one has to characterize first the nature of its traffic, QoS requirements, and any additional components or devices. For simplicity, we assume a point-to-point conversation for all VoIP calls with no call conferencing. For deploying VoIP, a *gatekeeper* or *CallManager* node has to be added to the network [3,4,5]. The *gatekeeper* node handles signaling for establishing, terminating, and authorizing connections of all VoIP calls. Also a VoIP *gateway* is required to handle external calls. A VoIP *gateway* is responsible for converting VoIP calls to/from the Public Switched Telephone Network (PSTN). As an engineering and design issue, the placement of these nodes in the network becomes crucial. We will tackle this issue in design step 5. Other hardware requirements include a VoIP client terminal, which can be a separate VoIP device, i.e., IP phones, or a typical PC or workstation that is VoIP-enabled. A VoIP-enabled workstation runs VoIP software such as IP SoftPhones [6-8].

Figure 3 identifies the end-to-end VoIP components from sender to receiver [9]. The first component is the *encoder* which periodically samples the original voice signal and assigns a fixed number of bits to each sample, creating a constant bit rate stream. The traditional sample-based encoder G.711 uses Pulse Code Modulation (PCM) to generate 8-bit samples every 0.125 ms, leading to a data rate of 64 kbps [10]. The *packetizer* follows the *encoder* and encapsulates a certain number of speech samples into packets and adds the RTP, UDP, IP, and Ethernet headers. The voice packets travel through the data network. An important component at the receiving end, is the *playback buffer* whose purpose is to absorb variations or jitter in delay and provide a smooth playout. Then packets are delivered to the *depacketizer* and eventually to the *decoder* which reconstructs the original voice signal.

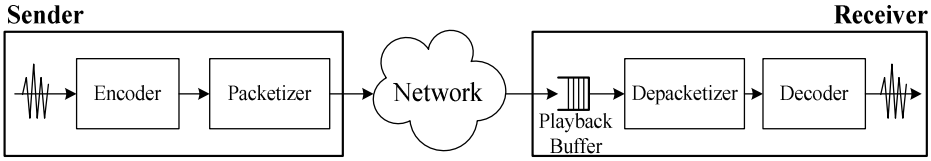


Figure 3. VoIP end-to-end components

We will follow the widely-adopted recommendations of H.323, G.711, and G.714 standards for VoIP QoS requirements [11-12]. Table 1 compares some commonly-used ITU-T standard codecs and the amount of one-way delay that they impose. To account for upper limits and to meet desirable quality requirement according to ITU recommendation P.800 [13], we will adopt G.711u codec standards for the required delay and bandwidth. G.711u yields around 4.4 MOS rating. MOS, *Mean Opinion Score*, is a commonly used VoIP performance metric given in a scale of 1 to 5, with 5 is the best [14,15]. However, with little compromise to quality, it is possible to implement different ITU-T codecs that yield much less required bandwidth per call and relatively a bit higher, but acceptable, end-to-end delay. This can be accomplished by applying compression, silence suppression, packet loss concealment, queue management techniques, and encapsulating more than one voice packet into a single Ethernet frame [3,9,16-21].

Table 1. Common ITU-T codecs and their defaults

Codec	Data rate (kbps)	Datagram size (ms)	A/D Conversion delay (ms)	Combined bandwidth (bi-directional) (kbps)
G.711u	64.0	20	1.0	180.80
G.711a	64.0	20	1.0	180.80
G.729	8.0	20	25.0	68.80
G.723.1 (MPMLQ)	6.3	30	67.5	47.80
G.723.1 (ACELP)	5.3	30	67.5	45.80

3.1.1 End-to-End Delay for a Single Voice Packet

Figure 3 illustrates the sources of delay for a typical voice packet. The end-to-end delay is sometimes referred to by M2E or Mouth-to-Ear delay [7]. G.714 imposes a maximum total one-way packet delay of 150ms end-to-end for VoIP applications [12]. In [22] a delay of up to 200ms was considered to be acceptable. We can break this delay down into at least three different contributing components, which are as follows (i) encoding, compression, and packetization delay at the sender (ii) propagation, transmission and queuing delay in the network and (iii) buffering, decompression, depacketization, decoding, and playback delay at the receiver.

3.1.2 Bandwidth for a Single Call

The required bandwidth for a single call, one direction, is 64 kbps. G.711 codec samples 20ms of voice per packet. Therefore, 50 such packets need to be transmitted per second. Each packet contains 160 voice samples in order to give 8000 samples per second. Each packet is sent in one Ethernet frame. With every packet of size 160 bytes, headers of additional protocol layers are added. These headers include RTP + UDP + IP + Ethernet

with preamble of sizes 12 + 8 + 20 + 26, respectively. Therefore, a total of 226 bytes, or 1808 bits, needs to be transmitted 50 times per second, or 90.4 kbps, in one direction. For both directions, the required bandwidth for a single call is 100 pps or 180.8 kbps assuming a symmetric flow.

3.1.3 Other Assumptions

Throughout our analysis and work, we assume voice calls are symmetric and no voice conferencing is implemented. We also ignore the signaling traffic generated by the *gatekeeper*. We base our analysis and design on the worst-case scenario for VoIP call traffic. The signaling traffic involving the *gatekeeper* is mostly generated prior to the establishment of the voice call and when the call is finished. This traffic is relatively small compared to the actual voice call traffic. In general, the *gatekeeper* generates no or very limited signaling traffic throughout the duration of the VoIP call for an already established on-going call [3].

In this paper, we will implement no QoS mechanisms that can enhance the quality of packet delivery in IP networks. A myriad of QoS standards are available and can be enabled for network elements. QoS standards may include IEEE 802.1p/Q, the IETF's RSVP, and DiffServ. Analysis of implementation cost, complexity, management, and benefit must be weighed carefully before adopting such QoS standards. These standards can be recommended when the cost for upgrading some network elements are high and the network resources are scarce and heavily loaded.

3.2 VoIP Traffic Flow and Call Distribution

Knowing the current telephone call usage or volume of the enterprise is an important step for a successful VoIP deployment. Before embarking on further analysis or planning phases for a VoIP deployment, collecting statistics about of the present call volume and profiles is essential. Sources of such information are organization's PBX, telephone records and bills. Key characteristics of existing calls can include the number of calls, number of concurrent calls, time, duration, etc. It is important to determine the locations of the call endpoints, i.e., the sources and destinations, as well as their corresponding path or flow. This will aid in identifying the call distribution and the calls made internally or externally. Call distribution must include percentage of calls within and outside of a floor, building, department, or organization. As a good capacity planning measure, it is recommended to base the VoIP call distribution on the busy hour traffic of phone calls for the busiest day of a week or a month. This will ensure support of the calls at all times with high QoS for all VoIP calls. When such current statistics are combined with the projected extra calls, we can predict the worst-case VoIP traffic load to be introduced to the existing network. Figure 4 describes the call distribution for the enterprise under study based on the worst busy hour and the projected future growth of VoIP calls. In the figure, the call distribution is described as a probability tree. It is also possible to describe it as a probability matrix.

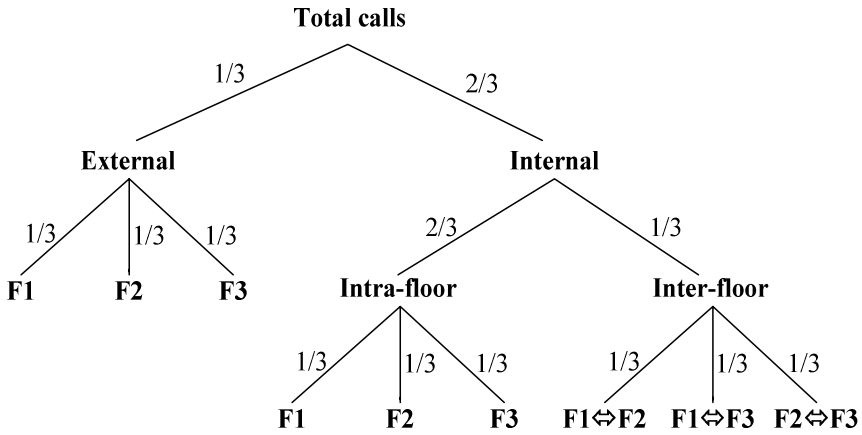


Figure 4. Probability tree describing the VoIP call distribution

Some important observations can be made about the voice traffic flow for inter-floor and external calls. For all these type of calls, the voice traffic has to be always routed through the router. This is so because Switch 1 and Switch 2 are layer 2 switches with VLANs configuration. One can observe that the traffic flow for inter-floor calls between Floor 1 and Floor 2 imposes twice the load on Switch 1, as the traffic has to pass through the switch to the router and back to the switch again. Similarly, Switch 2 experiences twice the load for external calls from/to Floor 3.

3.3 Define Performance Thresholds and Growth Capacity

In this step we define the network performance thresholds or operational points for a number of important key network elements. These thresholds are to be considered when deploying the new service. The benefit is twofold. First, the requirements of the new service to be deployed are satisfied. Second, adding the new service leaves the network healthy and susceptible to future growth.

Two important performance criteria are to be taken into account. First is the maximum tolerable end-to-end delay; and second is the utilization bounds or thresholds of network resources. The maximum tolerable end-to-end delay is determined by the most sensitive application to run on the network. In our case, it is 150ms end-to-end for VoIP. It is imperative to note that if the network has certain delay-sensitive applications, the delay for these applications should be monitored, when introducing VoIP traffic, such that they do not exceed their required maximum values. As for the utilization bounds for network resources, such bounds or thresholds are determined by factors such as current utilization, future plans, and foreseen growth of the network. Proper resource and capacity planning is crucial. Savvy network engineers must deploy new services with scalability in mind, and ascertain that the network will yield acceptable performance under heavy and peak loads, with no packet loss. VoIP requires almost no packet loss. In literature 0.1% to 5% packet loss was generally asserted [6,21-23]. However, in [24] the required VoIP packet loss was conservatively suggested to be less than 10^{-5}. A more practical packet loss, based on experimentation, of below 1% was required in [22]. Hence, it is extremely important not to utilize fully the network resources. As rule-of-thumb guideline for switched fast full-duplex Ethernet, the average utilization limit of links should be 190%, and for switched shared fast Ethernet, the average limit of links should be 85% [25].

The projected growth in users, network services, business, etc. must be all taken into consideration to extrapolate the required growth capacity or the future growth factor. In our study we will ascertain that 25% of the available network capacity is reserved for future growth and expansion. For simplicity, we will apply this evenly to all network resources of the router, switches, and switched-Ethernet links. However, keep in mind this percentage in practice can be variable for each network resource and may depend on the current utilization and the required growth capacity. In our methodology, the reservation of this utilization of network resources is done upfront, before deploying the new service, and only the left-over capacity is used for investigating the network support of the new service to be deployed.

3.4 Perform Network measurements

In order to characterize the existing network traffic load, utilization, and flow, network measurements have to be performed. This is a crucial step as it can potentially affect results to be used in analytical study and simulation. There are a number of tools available commercially and non-commercially to perform network measurements. Popular open-source measurement tools include MRTG, STG, SNMPUtil, and GetIF [26]. A few examples of popular commercially measurement tools include HP OpenView, Cisco Netflow, Lucent VitalSuite, Patrol DashBoard, Omegon NetAlly, Avaya ExamiNet, NetIQ Vivinet Assessor, etc.

Network measurements must be performed for network elements such as routers, switches, and links. Numerous types of measurements and statistics can be obtained using measurement tools. As a minimum, traffic rates in bps (bits per second) and pps (packets per second) must be measured for links directly connected to routers and switches. To get adequate assessment, network measurements have to be taken over a long period of time, at least 24-hour period. Sometimes it is desirable to take measurements over several days or a week.

Table 2. Worst-case network measurements

Link	Bit rate (Mbps)	Packet rate (pps)	Utilization
Router ⇔ Switch 1	9.44	812	9.44 %
Router ⇔ Switch 2	9.99	869	9.99 %
Switch 1 ⇔ Floor 1	3.05	283	6.1 %
Switch 1 ⇔ Floor 2	3.19	268	6.38 %
Switch 1 ⇔ File Server	1.89	153	1.89 %
Switch 1 ⇔ DB Server	2.19	172	2.19 %
Switch 2 ⇔ Floor 3	3.73	312	7.46 %
Switch 2 ⇔ Email Server	2.12	191	2.12 %
Switch 2 ⇔ HTTP Server	1.86	161	1.86 %
Switch 2 ⇔ Firewall	2.11	180	2.11 %
Switch 2 ⇔ Proxy	1.97	176	1.97 %

One has to consider the worst-case scenario for network load or utilization in order to ensure good QoS at all times including peak hours. The peak hour is different from one network to another and it depends totally on the nature of business and the services provided by the network. Table 2 shows a summary of peak-hour

utilization for traffic of links in both directions connected to the router and the two switches of the network topology of Figure 1. These measured results will be used in our analysis and simulation study.

3.5 Upfront Network Assessment and Modifications

In this step we assess the existing network and determine, based on the existing traffic load and the requirements of the new service to be deployed, if any immediate modifications are necessary. Immediate modifications to the network may include adding and placing new servers or devices, upgrading PCs, and re-dimensioning heavily utilized links. As a good upgrade rule, topology changes need to be kept to minimum and should not be made unless it is necessary and justifiable. Over-engineering the network and premature upgrades are costly and considered as poor design practices.

Based on the existing traffic load discussed in design step 4 of Section 3.4, all the links connecting the router and the switches and links connecting the servers and the switches are underutilized. If any of the links were heavily utilized, e.g. 30-50%, the network engineer should decide to re-dimension the link to 1-Gbps link at this stage. As for shared links of Floor 1, Floor 2, and Floor 3, the replacement or re-dimensioning of these links must be decided on carefully. At first, it looks costly effective not to replace the shared-Ethernet LAN for each floor with a switched LAN. However shared Ethernet scales poorly. More importantly, shared Ethernet offers zero QoS and are not recommended for real-time and delay-sensitive applications as it introduces excessive and variable latency under heavy loads and when subjected to intense bursty traffic [25]. In order to consistently maintain the VoIP QoS, a switched fast full-duplex Ethernet LAN becomes necessary.

Based on the hardware requirement for deploying VoIP described in Section 3.1, two new nodes have to be added to the existing network: a VoIP *gateway* and a *gatekeeper*. As a network design issue, an appropriate node placement is required for these two nodes. Since most of the users reside on Floor 1 and Floor 2 and connected directly to Switch 1, connecting the *gatekeeper* to Switch 1 is practical in order to keep the traffic local. For the VoIP *gateway*, we connect it to Switch 2 in order to balance the projected load on both switches. Also it is more reliable and fault-tolerant not to connect both nodes to the same switch in order to eliminate problems that stem from a single point of failure. For example, if Switch 2 fails, only external calls to/from the network will be affected. It is proper to include the gatekeeper to be a member of VLAN1 of Switch 1 which includes the database and file servers. This isolates the gatekeeper from multicast and broadcast traffic of Floor 1 and Floor 2. In addition, the *gatekeeper* can access locally the database and file servers to record and log phone calls. On the other hand, we create a separate VLAN for the *gateway* in order to isolate the *gateway* from multicast and broadcast traffic of Floor 3 and the servers of switch 2. Therefore, the network has now a total of six VLANs.

Figure 5 shows the new network topology after the incorporation of necessary VoIP components. As shown, two new *gateway* and *gatekeeper* nodes for VoIP were added and the three shared Ethernet LANs were replaced by 100Mbps switched Ethernet LANs.

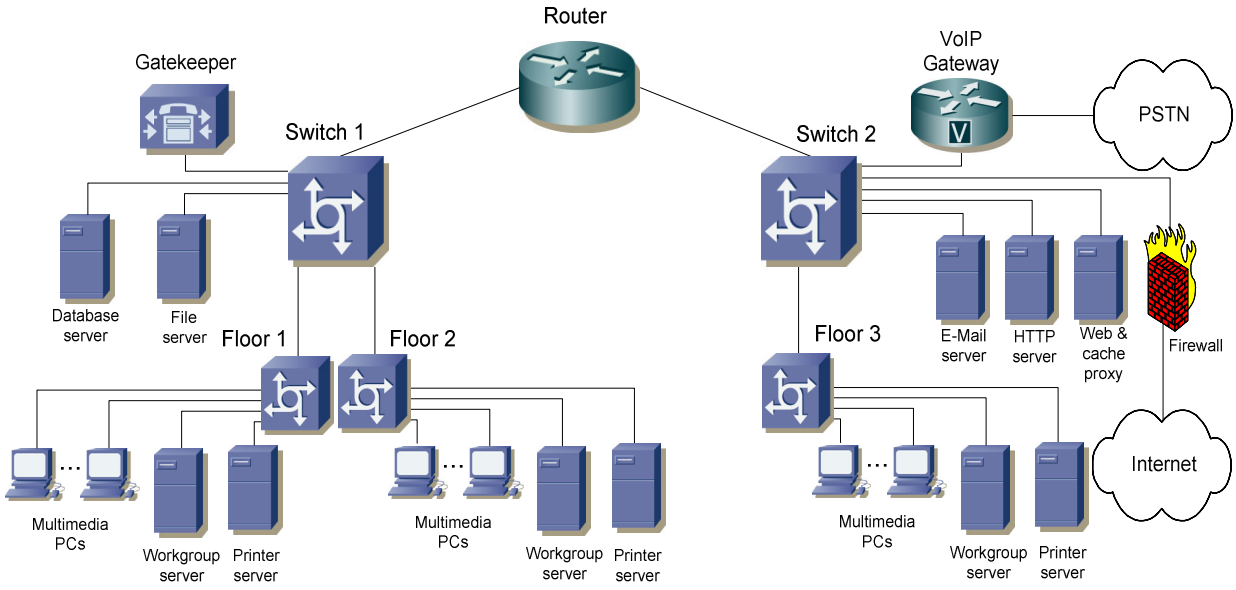


Figure 5. Network topology with VoIP Components

3.6 Analysis

VoIP is bounded by two important metrics. First is the available bandwidth. Second is the end-to-end delay. The actual number of VoIP calls that the network can sustain and support is bounded by those two metrics. Depending on the network under study, either the available bandwidth or delay can be the key dominant factor in determining the number of calls that can be supported.

3.6.1 Bandwidth Bottleneck Analysis

Bandwidth bottleneck analysis is an important step to identify the network element, whether it is a node or a link, that puts a limit on how many VoIP calls can be supported by the existing network. For any path that has N network nodes and links, the bottleneck network element is the node or link that has the minimum available bandwidth. According to [27], this minimum available bandwidth is defined as follows

$$A = \min_{i=1,\ldots,N} A_i,$$

and

$$A_i = (1-u_i)C_i,$$

where C_i is the capacity of network element i and u_i is its current utilization. The capacity C_i is the maximum possible transfer or processing rate.

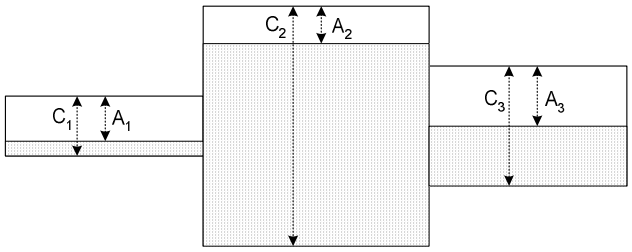


Figure 6. Bandwidth bottleneck for a path of three network elements

Therefore the theoretical maximum number of calls that can be supported by a network element E_i can be expressed in terms of A_i as

$$MaxCalls_i = \frac{A_i(1-growth_i)}{CallBW}, \tag{1}$$

where $growth_i$ is the growth factor of network element E_i, and takes a value from 0 to 1. *CallBW* is the VoIP bandwidth for a single call imposed on E_i. As previously discussed in design step 2 of Section 3.2, the bandwidth for one direction is given as 50 pps or 90.4 kbps. In order to find the bottleneck network element that limits the total number of VoIP calls, one has to compute the maximum number of calls that can be supported by each network element, as in equation (1), and the percentage of VoIP traffic flow passing by this element. The percentage of VoIP traffic flow for E_i, denoted as $flow_i$, can be found by examining the distribution of the calls. The total number of VoIP calls that can be supported by a network can be expressed as

$$TotalCallsSupported = \min_{i=1,\ldots,N}\left(\frac{MaxCalls_i}{flow_i}\right). \tag{2}$$

Let us for the sake of illustration compute the $MaxCalls_i$ and $flow_i$ supported by the Router, Switch 1, and uplink from Switch 2 to the Router. Table 3 shows the maximum calls that can be supported by those network elements. For our network example, we choose $growth_i$ to be 25% for all network elements. u_i is determined by Table 2. C_i, for the router and the switch is usually given by the product datasheets. According to [28] and [29], the capacity C_i for the router or the switch, is 25,000pps and 1.3M pps, respectively. $flow_i$ is computed by examining the probability tree for call distribution shown in Figure 4.

Table 3. Maximum VoIP calls support for few network elements

Network Element	C_i	u_i	*CallBW*	$flow_i$	$MaxCalls_i$
Router	25,000 pps	6.72%	100 pps	5/9	174
Switch 1	1.3 Mpps	0.13%	100 pps	14/27	9,737
Uplink from Switch 2 to Router	100 Mbps	9.99%	90.4 kbps	16/27	746

Table 3 shows the $MaxCalls_i$ for only three network elements. In order to find the actual calls that the network can sustain, i.e. *TotalCallsSupported* of equation (2), $flow_i$ and $MaxCalls_i$ have to be computed for all

network elements. This can be automated by implementing the equations using MATLAB, and therefore these values can be computed quickly. When computing the $MaxCalls_i$ for all network elements, it turns out that the router is the bottleneck element. Hence, *TotalCallsSupported* is 313 VoIP calls.

For the sake of illustration, we show how u_i and $flow_i$ can be computed. u_i can be computed by Table 2. For example, the utilization for the router is the total incoming traffic (or received traffic) into the router divided by the router's capacity. According to Table 2, this yields to (812+869)/25000=6.72%. $flow_i$ can be computed using the probability tree shown in Figure 4 as follows. For the router, $flow_i$ is the percentage of the inter-floor and external calls, which is (2/3)(1/3)+1/3. Similarly, $flow_i$ for Switch 1 and the uplink from Switch 2 to the router would be 14/27 and 16/27, respectively. For Switch 1, $flow_i$ is the percentage of external calls going out of Floor 1 and Floor 2, plus the percentage of inter-floor calls between Floor 1 and Floor 2, Floor 1 and Floor 3, and Floor 2 and Floor 3. This can be expressed as (1/3){1/3+1/3} + (2/3)(1/3){2/3 +(1/3)(1/3)}. Note that the fraction of inter-floor calls between Floor 1 and Floor 2 is 2/3 since the calls pass through the switch twice as they get routed by the router back to Switch 1. See Section 3.2. For the uplink from Switch 2 to the router, $flow_i$ is the percentage of external calls going out of the three floors plus the percentage of inter-floor calls between Floor 1 and Floor3 and Floor 2 and Floor 3. This can be simply expressed as (1/3){1/3+1/3+2/3} + (2/3)(1/3){1/3+1/3}. Note that the fraction of the external calls going out of Floor 3 is 2/3 since the calls pass through the link twice as they get routed by the router.

3.6.2 Delay Analysis

As defined in Section 3.3 for the existing network, the maximum tolerable end-to-end delay for a VoIP packet is 150 ms. The maximum number of VoIP calls that the network can sustain is bounded by this delay. We must always ascertain that the worst-case end-to-end delay for all the calls must be less than 150 ms. It should be kept in mind that our goal is to determine the network capacity for VoIP, i.e. the maximum number of calls that existing network can support while maintaining VoIP QoS. This can be done by adding calls incrementally to the network while monitoring the threshold or bound for VoIP delay. When the end-to-end delay, including network delay, becomes larger than 150 ms, the maximum number of calls can then be known.

As described in Section 3.1.1, there are three sources of delay for a VoIP stream: sender, network, and receiver. An equation is given in [24] to compute the end-to-end delay *D* for a VoIP flow in one direction from sender to receiver.

$$D = D_{pack} + \sum_{h \in Path} (T_h + Q_h + P_h) + D_{play},$$

where D_{pack} is the delay due to packetization at the source. At the source, there is also D_{enc} and $D_{process}$. D_{enc} is the encoder delay of converting A/D signal into samples. $D_{process}$ is the PC of IP phone processing that includes encapsulation . In G.711, D_{pack} and D_{enc}, are 20 ms and 1ms, respectively. Hence, it is appropriate for our analysis to have a fixed delay of 25 ms being introduced at the source, assuming worst case situation.

D_{play} is the playback delay at the receiver, including jitter buffer delay. The jitter delay is at most 2 packets, i.e. 40ms. If the receiver's delay of $D_{process}$ is added, we obtain a total fixed delay of 45 ms at the receiver. $T_h + Q_h + P_h$ is the sum of delays incurred in the packet network due to transmission, queuing, and propagation going through each hop *h* in the path from the sender to the receiver. The propagation delay P_h is typically ignored for traffic within a LAN, but not for a WAN. For transmission delay T_h and queueing delay Q_h we apply queueing theory. Hence the delay to be introduced by the network, expressed as $\sum_{h \in Path} (T_h + Q_h)$, should not exceed (150 – 25 – 45) or 80 ms.

We utilize queueing analysis to approximate and determine the maximum number of calls that the existing network can support while maintaining a delay of less than 80ms. In order to find the network delay, we utilize the principles of Jackson theorem for analyzing queueing networks. In particular, we use the approximation method of analyzing queueing networks by decomposition discussed in [30]. In this method, the arrival rate is assumed to be Poisson and the service times of network elements are exponentially distributed. Analysis by decomposition is summarized in first isolating the queueing network into subsystems, e.g., single queueing node. Next, analyzing each subsystem separately, considering its own network surroundings of arrivals and departures. Then, finding the average delay for each individual queueing subsystem. And finally, aggregating all the delays of queueing subsystems to find the average total end-to-end network delay.

For our analysis we assume the VoIP traffic to be Poisson. In reality, the inter-arrival time, $1/\lambda$, of VoIP packets is constant, and hence its distribution is deterministic. However, modeling the voice arrival as Poisson gives adequate approximation according to [24], especially when employing a high number of calls. More importantly, the network element with a non-Poisson arrival rate makes it difficult to approximate the delay and lead to intractable analytical solution. Furthermore, analysis by decomposition method will be violated if the arrival rate is not Poisson.

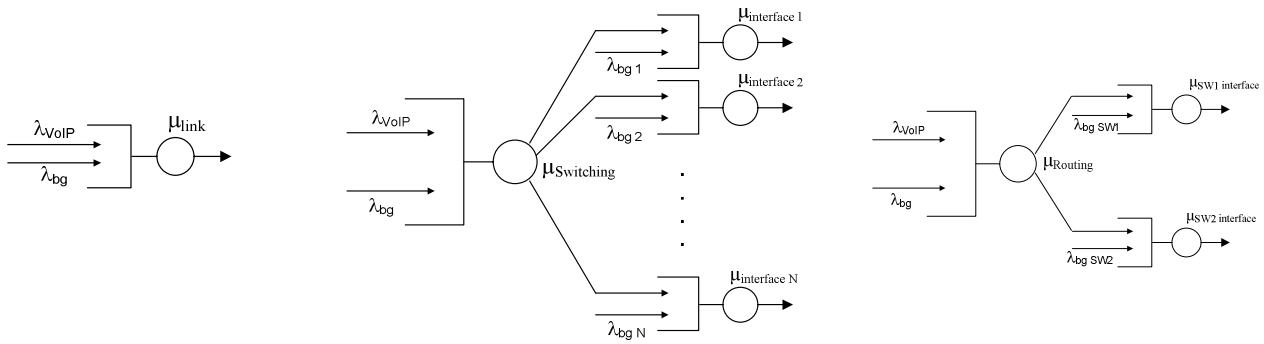


Figure 7. Queueing models for three network elements

Figure 7 shows queueing models for three network elements of the router, switch and link. The queueing model for the router has two outgoing interfaces: an interface for SW1 and another for SW2. The number of

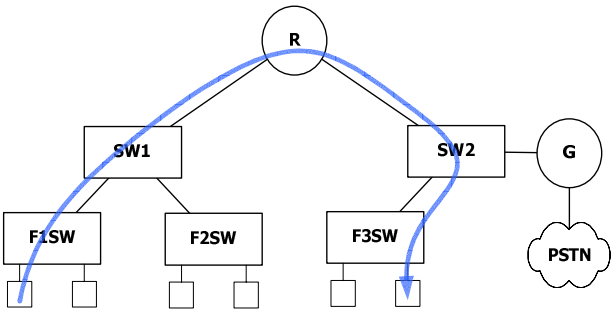
outgoing interfaces for the switches are many, and such a number depends on the number of ports for the switch. We modeled the switches and the router as *M/M/1* queues. Ethernet links are modeled as *M/D/1* queues. This is appropriate since the service time for Ethernet links is more of a deterministic than variable. However, the service times of the switches and the router are not deterministic since these are all CPU-based devices. According to the datasheet found in [28,29], the switches and the router used in Figure 1 have somewhat similar design of a store-and-forward buffer pool with a CPU responsible for pointer manipulation to switch or route a packet to different ports. [31] provides a comprehensive models of common types of switches and routers. According to [32], the average delay for a VoIP packet passing through an *M/M/1* queue is basically $1/(\mu-\lambda)$, and through an *M/D/1* queue is $\left(1-\frac{\lambda}{2\mu}\right)/(\mu-\lambda)$, where λ is the mean packet arrival rate and μ is the mean network element service rate. The queueing models in Figure 7 assume Poisson arrival for both VoIP and background traffic. In [24], it was concluded that modeling VoIP traffic as Poisson is adequate. However and in practice, background traffic is bursty in nature and characterized as self-similar with long range dependence [33]. For our analysis and design, using bursty background traffic is not practical. For one thing, under the network of queues being considered an analytical solution becomes intractable when considering non-Poisson arrival. Also, it is important to remember that in order to ensure good QoS at all times, we base our analysis and design on the worst-case scenario of network load or utilization, i.e., the peak of aggregate bursts. And thus in a way our analytical approach takes into account the bursty nature of traffic.

It is worth noting that the analysis by decomposition of queueing networks in [30] assumes exponential service times for all network elements including links. But [34] proves that acceptable results with adequate accuracy can be still obtained if the homogeneity of service times of nodes in the queueing network is deviated. [34] shows that the main system performance is insensitive to violations of the homogeneity of service times. Also, it was noted that when changing the models for links from *M/D/1* to *M/M/1*, a negligible difference was observed. More importantly, as will be demonstrated in this paper with simulation, our analysis gives a good approximation.

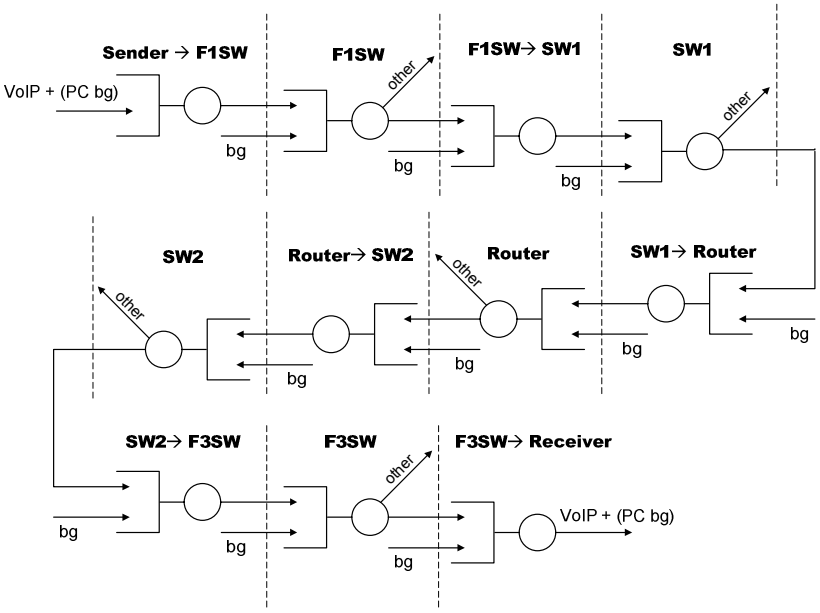
The total end-to-end network delay starts from the Ethernet outgoing link of the sender PC or IP phone to the incoming link of receiver PC or IP phone. To illustrate this further, let us compute the end-to-end delay encountered for a single call initiated from Floor 1 to Floor 3. Figure 8 shows an example of how to compute the network delay. Figure 8a shows the path of a unidirectional voice traffic flow going from Floor 1 to Floor 3. Figure 8b shows the corresponding networking queueing model for such a path.

For Figure 8b, in order to compute the end-to-end delay for a single bi-directional VoIP call, we must compute the delay at each network element. We show how to compute the delay for the switches, links, and router. For the switch, whether it is that of intra-floor or inter-floor, $\mu=(1-25\%)\times 1.3$ Mpps, where 25% is the growth factor. $\lambda=\lambda_{VoIP}+\lambda_{bg}$, where λ_{VoIP} is the total added new traffic from a single VoIP in pps, and λ_{bg} is the background traffic in pps. For an uplink or downlink, $\mu=(1-25\%)\times 100$Mbps, $\lambda=\lambda_{VoIP}+\lambda_{bg}$. Since the service rate is in bps, λ_{VoIP} and λ_{bg} must be expressed in bps. Table 2 and Table 3 express the bandwidth for

background traffic and for a single call in both pps and bps. Similarly for the router, $\mu = (1-25\%)\times 25{,}000$pps and $\lambda = \lambda_{VoIP} + \lambda_{bg}$. Both λ_{VoIP} and λ_{bg} must be expressed in pps. Remember for a single bi-directional VoIP call, λ_{VoIP} at the router and switches for a single call will be equal to 100pps. However, for the uplink and downlink links, it is 90.4 kbps. One should consider no λ_{bg} for the outgoing link if IP phones are used. For multimedia PCs which equipped with VoIP software, a λ_{bg} of 10% of the total background traffic is utilized in each floor. For a more accurate assessment of PC's λ_{bg}, actual measurement should be taken. For our case study of the small enterprise network, we use multimedia PCs.



(a) Unidirectional voice traffic flow path from Floor 1 to Floor 3



(b) Corresponding network queueing model of the entire path

Figure 8. Computing network delay

The total delay for a single VoIP call of Figure 8b, can be determined as follows:

$$D_{path} = D_{Sender-F1SW\ Link} + D_{F1SW} + D_{F1SW-SW1\ Link} + D_{SW1} + D_{SW1-Router\ Link} + D_{Router} + D_{Router-SW2\ Link} + D_{SW2} + D_{SW2-F3SW\ Link} + D_{F3SW} + D_{F3SW-Receiver\ Link}$$

Network Capacity Algorithm. In order to determine the maximum number of calls that can be supported by an existing network while maintaining VoIP delay constraint, we developed the following algorithm that basically determines network capacity in terms of VoIP calls. Calls are added iteratively until the worst-case network delay of 80 ms has reached. The algorithm can be described in the following steps:

i) Initially, no calls are introduced and the only traffic in the network is the background traffic.
ii) A new call is added, according to the call distribution described in Figure 4.
iii) For each network element, $\lambda = \lambda_{VoIP} + \lambda_{bg}$ is computed. λ_{bg} is known for each element; however, λ_{VoIP} can get affected by introducing a new call depending on the call traffic flow, i.e. whether or not the new call flow passes through the network element.
iv) For each network element, the average delay of a VoIP packet is computed.
v) The end-to-end delay is computed by summing up all the delays of step (iv) encountered for each possible VoIP flow. This includes all external and internal flows, with internal flows consisting of intra-floor and inter-floor.
vi) The maximum network delay of all possible flows is determined. If the maximum network delay is less than 80 ms, then the maximum number of calls has not been reached. Therefore a new call can be added, and hence go to step (ii).
vii) If not, the maximum delay has been reached. Therefore the number of VoIP calls bounded by the delay is one less than the last call addition.

The above algorithm was implemented using MATLAB and the results for the worst incurred delay are plotted in Figure 9. It can be observed from the figure that the delay increases sharply when the number of calls go beyond 310 calls. To be more precise, MATLAB results showed the number of calls that are bounded by the 80 ms delay is 316.

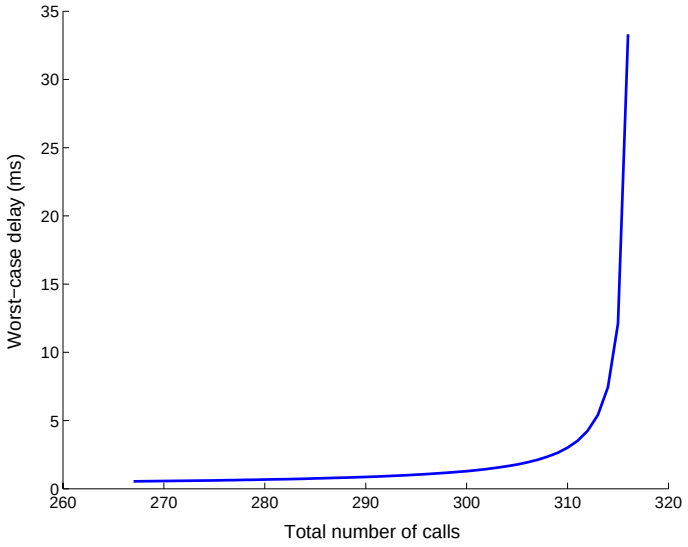


Figure 9. Worst incurred delay vs. number of VoIP calls

When comparing the number of calls that network can sustain based on bottleneck bandwidth and worst-delay analysis, we find the number of calls is limited by the available bandwidth more than the delay, though the difference is small. Therefore, we can conclude that the maximum number of calls that can be sustained by the existing network is 313.

Packet Loss. A question related to determining the number of calls to be supported by a particular data network is packet loss. VoIP packet loss should be below 1% according to [22], and hence packet loss can be a third constraint that plays a key role in determining the number of calls to be supported by a network. In this case, finite queueing systems of *M/M/1/B* and *M/D/1/B*, as opposed to *M/M/1* and *M/D/1*, must be used instead. In a finite queueing system, due to dropping of packets, the flow of one node will affect the flow of another because we have bidirectional flows. Consequently, we end up with a model of somewhat closed queueing networks with blocking [35]. Determining packet loss for this type of networks is not a trivial task, and can be only approximated, according to [35,36]. Approximation algorithms found in literature for solving closed networking queueing systems are not accurate and does not have a closed form solution. The solution is typically heuristic and it takes a long time to converge [35]. Due to lack of closed-form analytical solutions and according to [36], simulation is a more practical approach to study packet loss. In the work presented in this paper, we will use simulation to verify that the packet loss constraint is satisfied with no packet loss.

3.7 Simulation

The object of the simulation is to verify analysis results of supporting VoIP calls. We used the popular MIL3's OPNET Modeler simulation package[1], Release 8.0.C [37]. OPNET Modeler contains a vast amount of models of commercially available network elements, and has various real-life network configuration capabilities. This makes the simulation of real-life network environment close to reality. Other features of OPNET include GUI interface, comprehensive library of network protocols and models, source code for all models, graphical results and statistics, etc. More importantly, OPNET has gained considerable popularity in academia as it is being offered free of charge to academic institutions. That has given OPNET an edge over DES NS2 in both market place and academia. This section gives a brief description of the simulation model, configurations, and results.

3.7.1 Modeling the Network

A snapshot of the OPNET simulation model for the existing network under study is shown in Figure 10. The simulation model of the organization network, for the most part, is an exact replica of the real network. In OPNET Modeler, many vendor-specific models are included in the pre-defined component libraries. VoIP *gateway* is modeled as an Ethernet workstation. The enterprise servers are modeled as Ethernet servers. All network elements have been connected using a 100 Base-T links. Figure 10 shows the described topology. As discussed in Section 3.1.3, the *gatekeeper* signaling traffic is ignored, and hence modeling such and element and its traffic is not taken into account as we base our study on the worst case situation.

[1] OPNET Modeler was provided under the OPNET University Programs

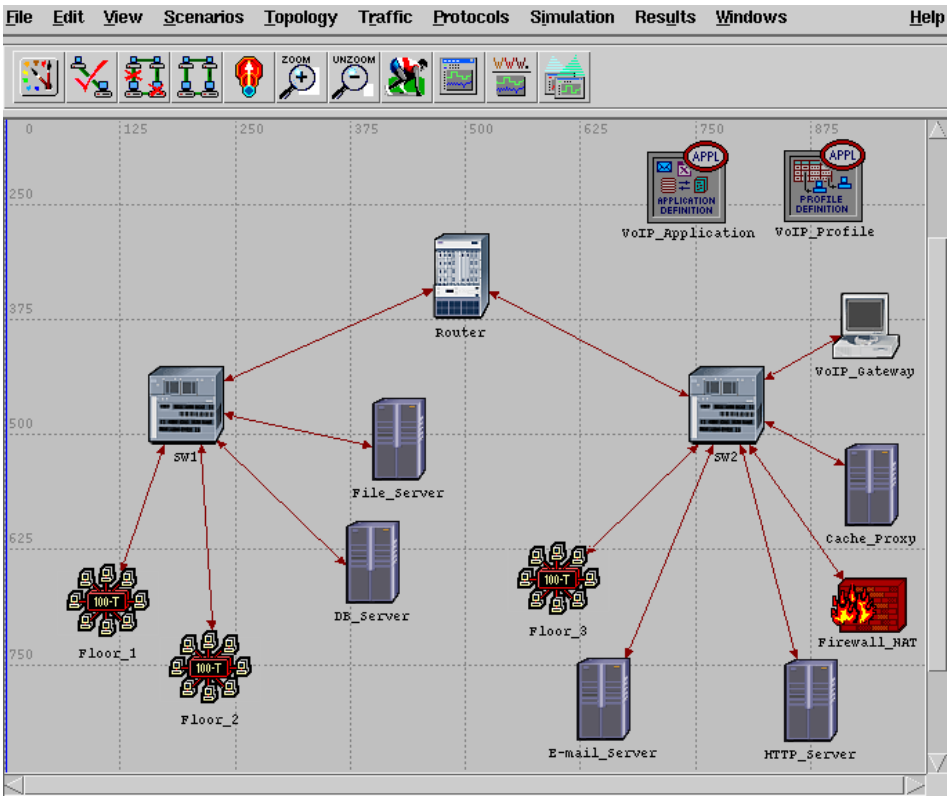


Figure 10. The organization network topology

Floor LANs have been modeled as subnets that enclose an Ethernet switch and three Ethernet workstations used to model the traffic of the LAN users, as shown in Figure 11. One of these workstations generates the background traffic of the floor while the other two act as parties in VoIP sessions. For example, the Ethernet workstations for Floor 1 are labeled as `F1_C2`, `F1_C2`, and `F1_C3`. `F1_C1` is the source for sending VoIP calls. `F1_C2` is the sink for receiving VoIP calls. `F1_C3` is the sink and source of background traffic.

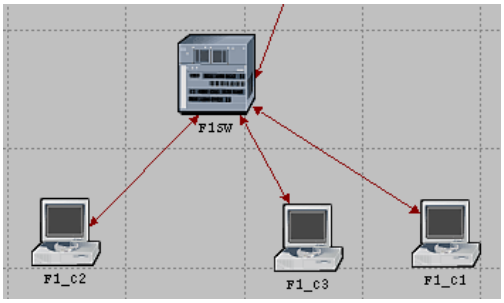


Figure 11. Floor 1 subnet model

Various OPNET Modeler configurations were made which included the network VLANs, router, switches, and links. Also background traffic was incorporated into the network as well as the generation of VoIP traffic. For VoIP traffic generation, a VoIP application and a profile have to be created. OPNET Modeler has a predefined voice application. The VoIP traffic got generated and received by workstations within the floors. The VoIP

traffic was generated according to the flow and call distribution discussed in Section 3.2. We set up OPNET Modeler such that three new VoIP calls are generated every two seconds.

3.7.2 Simulation Results

In this section, the most relevant graphed results for the VoIP traffic volume and delay are reported. We configured the duration of the OPNET simulation run for 8 minutes. The generation of background traffic, by default in OPNET, started at 40 seconds from the start time of the simulation run. The VoIP traffic started at 70 seconds at which a total of 3 VoIP bi-directional calls are initially added. Then, every 2 seconds 3 VoIP calls are added. The Simulation stops at 8 minutes in which a total of $3+\left(\frac{7\times 60+58-70}{2}\right)\times 3=615$ calls got generated. This should translate into a total of 61,500 packets being generated every second. Note that since the simulation stops at 8 minutes, the last 3 calls to be added was at 7 minutes and 58 seconds.

Figure 12 shows the VoIP traffic and the corresponding end-to-end delay as VoIP calls are added every two seconds. Figure 12a shows the total VoIP traffic that was sent, received, and dropped. Figure 12b is a zoom-in version of Figure 12a, focusing on the mismatch region between traffic sent and received. From Figure 12a, it is clear that the total VoIP traffic generated by the end of simulation run is very close 61,500 pps. In fact, simulation results gave 61,429 pps.

One can determine the total number of calls that the network can sustain by examining network bandwidth or delay bounds. We first investigate the bandwidth bound. Figure 12a and Figure 12b show clearly that not all of VoIP packets being sent get received. I.e. there is a mismatch between traffic sent and received. Figure 12b captures clearly the addition of the three calls every 2 seconds; and how this addition is repeated in gradual steps of 300 pps. We can determine the number of calls that can be supported by examining the X and Y axes. Examining the X axis of the simulation run time, it is clear that the last successful addition of three calls was at exactly 4 minutes and 48 seconds, as seen clearly in Figure 12b. The next addition, as shown, was at 4 minutes and 50 seconds which resulted in a mismatch. For the last successful addition of voice calls, which occurred at 4 minutes and 48 seconds, we had a traffic volume (see Y axis) of exactly 33,000 pps or 330 VoIP calls. Also one can arrive at the same number of calls by calculating how many calls have been added until the last successful addition of three calls, i.e. 4 minutes and 48 seconds. This yields to $3+\left(\frac{4\times 60+48-70}{2}\right)\times 3=330$ calls.

Figure 12c shows the corresponding VoIP end-to-end delay. Remember this delay should not exceed 80 ms, as discussed in Section 3.6.2. As depicted, the delay stays less than 80 ms until a simulation time of 4 minutes and 54 seconds at which the delay increases sharply. One can then find out the number of VoIP calls that the network can support to satisfy the 80 ms time constraint. The number of calls can be computed as $3+\left(\frac{4\times 60+54-70}{2}\right)\times 3=339$ calls. Therefore one can conclude that, based on these simulation results, the

number of voice calls to be supported by the network is bounded more by the network bandwidth than the delay. Hence, the number of the VoIP calls that the network can support based on simulation is 330 calls.

The simulation's reported delays shown in Figure 12c is the maximum values of a bucket of 100 collected values. The OPNET default reported delay configuration is the sample mean of a bucket of 100 collected values. Figure 12d depicts a different collection mode, in which "all values" are collected and plotted. Figure 12d depicts two types of delays. First, the delays of external and inter-floor VoIP packets passing through the router. These are the bigger delays and they resemble the delays of Figure 12c. Second, the delays of intra-floor VoIP packets that are not passing through the router. These are the smaller delays, in which the majority of these values stay close to 2.5 ms.

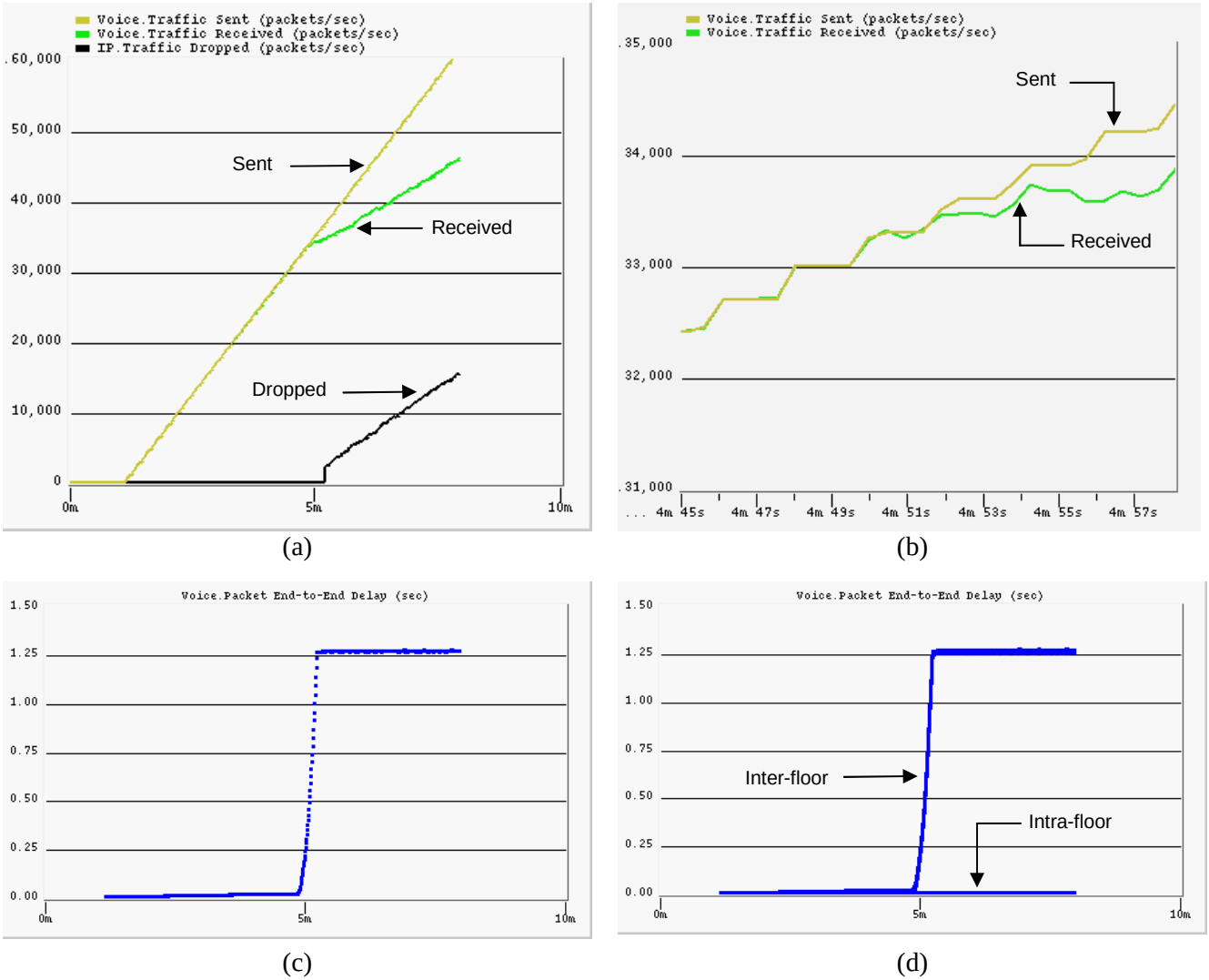


Figure 12. VoIP traffic and delay

When comparing the VoIP end-to-end delay in Figure 12c with the delay obtained by analysis in Figure 9, it is shown that the delay in Figure 12c does not shoot to infinity as that of Figure 9. In Figure 12c, the delay stays

flat at about 1.25 seconds. This is so because in our analysis we modeled the network elements with infinite buffer. Another observation can be made about the dropped VoIP packets in Figure 12a. We see that the dropping of VoIP packets occurs after the mismatch of the sent and received packets. This is due to the fact that CPU processing, especially of the router, gets 100% saturated before the memory buffer of the router gets filled up. It was observed that the memory buffer of the router gets completely full 25 seconds after the router's CPU utilization reaches 100%.

3.7.3 Simulation Accuracy

In order to gain accuracy (with a narrow confidence interval) of our simulation results, following the popular guidelines presented in [38,39], five simulation replications were run by feeding OPNET with different initial seeds. OPNET's pseudo random number generator is based on BSD's algorithm which permits safely, i.e., with no concern of overlapping of random number streams, any integer value to be an initial seed. Five simulation replications were sufficient. Each simulation replication produced very similar graphical results, which when interpreted as done in Section 3.7.2, led to the same total number of calls to be supported.

3.7.4 Final Simulation Run

Based on the simulation results, the existing network can support 330 VoIP calls. In our simulation, the voice calls were added every two seconds and the simulation was not allowed to stabilize for a long time. Our attention was focused on finding out the number of voice calls that the network can sustain. As a final check to ensure a healthy network and a normal behavior for all network elements, we perform a final simulation run in which 330 voice calls are added, all at once at the start of the simulation, say after 70 seconds. We let the simulation run execute for a prolong amount of time, say good 5 minutes, to reach a steady state. Then we examine the health of each network element. In our example, this final simulation of 330 voice calls was not successful. The simulation run showed a mismatch between traffic sent and received and a delay of more than 80 ms. However, a successful simulation run of 306 voice calls showed normal and healthy results with no packet loss, average delay of 2.15 ms, and adequate utilization of router and switch CPUs and links. Therefore we can conclude, based on OPNET simulation, that the network can support a total of 306 voice calls.

3.8 Pilot Deployment

Before embarking on changing any of the network equipment, it is always recommended to build a pilot deployment of VoIP in a test lab to ensure smooth upgrade and transition with minimum disruption of network services. A pilot deployment comes after training of IT staff. A pilot deployment is the place for the network engineers, support and maintenance team to get firsthand experience with VoIP systems and their behavior. During the pilot deployment, the new VoIP devices and equipment are evaluated, configured, tuned, tested, managed, monitored, etc. The whole team needs to get comfortable with how VoIP works, how it mixes with other traffic, how to diagnose and troubleshoot potential problems. Simple VoIP calls can be set up and some benchmark testing can be performed.

4 Design and Engineering Decisions

The following network design and engineering decisions can be justified from the analytic and simulation approaches:

- The existing network, with a reserved growth factor of 25%, can safely support up to 306 calls while meeting the VoIP QoS requirements and having no negative impact on the performance of existing network services or applications.
- For 306 calls, a network delay of about 2 ms is encountered. To be precise, analysis gave a delay of 1.50 ms, while simulation gave a delay of 2.15 ms.
- A safety growth factor of 25% is maintained across all network resources.
- The primary bottleneck of the network is the router. If the enterprise under study is expected to grow in the near future, i.e., more calls are required than 306 calls, the router replacement is a must. The router can be replaced with a popular Layer-3 Ethernet switch, and thus relieving the router from routing inter-floor calls from Floor 1 to Floor 2. Before prematurely changing other network components, one has to find out how many VoIP calls can be sustained by replacing the router. To accomplish this, the design steps and guidelines outlined in this paper must be revisited and re-executed.
- The network capacity to support VoIP is bounded more by the network throughput than the delay. This is due to the fact the existing network under study is small and does not have a large number of intermediate nodes. The network delay bound can become dominat if we a have a large-scale LAN or WAN.

5 Conclusion

The paper outlined a step-by-step methodology on how VoIP can be deployed successfully. The methodology can help network researchers and designers to determine quickly and easily how well VoIP will perform on a network prior to deployment. Prior to the purchase and deployment of VoIP equipment, it is possible to predict the number of VoIP calls that can be sustained by the network while satisfying QoS requirements of all existing and new network services and leaving enough capacity for future growth. In addition, the paper discussed many design and engineering issues pertaining to the deployment of VoIP. These issues include characteristics of VoIP traffic and QoS requirements, VoIP flow and call distribution, defining future growth capacity, and measurement and impact of background traffic.

We considered a case study of deploying VoIP in a small enterprise network. We applied the methodology and guidelines outlined in this paper on such a network. We utilized both analysis and simulation to determine the number of VoIP calls that can be supported for such a network. From results of analysis and simulation, it is apparent that both results are in line and give a close match. Based on the analytic approach, a total of 313 calls can be supported. Based on the simulation approach, a total of 306 calls can be supported There is only a difference of 7 calls. The difference can be contributed to the degree of accuracy between the analytic approach and OPNET simulation. Our analytic approach is an approximation. Also, the difference is linked to the way the OPNET Modeler adds the distribution of the calls. It was found that external and inter-floor calls

are added before intra-floor calls. In anyways, to be safe and conservative, one can consider the minimum number of calls of the two approaches.

In this paper, only peer-to-peer voice calls were considered. As a future work, one can consider implementing important VoIP options such as VoIP conferencing and messaging. Also as a future work, one can look into assessing the network support and readiness of deploying other popular real-time network services such multimedia, video, and web conferencing. As a near-term work, we are in the process of developing a GUI-based analytical design tool that automates the analytical approach presented in this paper in order to find the maximum number of VoIP calls that can supported by any given generic network topology.

Acknowledgements

The author acknowledges the support of King Fahd University of Petroleum and Minerals in the development of this work. Special thanks go to previous ICS Department graduate students (Mr. A. Alkhoraidly, Mr. M. Turki, Mr. R. Alghanmi and Mr. A. Alsanad). The technical work of Mr. A. Alkhoraidly has been greatly appreciated. The author also acknowledges the anonymous reviewers for their valuable comments on the earlier versions of this article.

References

[1] M. Bearden, L. Denby, B. Karacali, J. Meloche, and D. T. Stott, "Assessing Network Readiness for IP Telephony," Proceedings of IEEE International Conference on Communications, ICC02, vol.4, 2002, pp. 2568-2572

[2] B. Karacali, L. Denby, and J. Melche, "Scalable Network Assessment for IP Telephony," Proceedings of IEEE International Conference on Communications (ICC04), Paris, June 2004, pp. 1505-1511.

[3] Goode B, "Voice over Internet Protocol (VoIP)," Proceedings of IEEE, vol. 90, no. 9, Sept. 2002, pp. 1495-1517.

[4] P. Mehta and S. Udani, "Voice over IP", *IEEE Potentials Magazine*, vol. 20, no. 4, October 2001, pp. 36-40.

[5] W. Jiang and H. Schulzrinne, "Towards Junking the PBX: Deploying IP Telephony," Proceedings of ACM 11th International Workshop on Network and Operating System Support for Digital Audio and Video, Port Jefferson, NY, June 2001, pp. 177-185

[6] B. Duysburgh, S. Vanhastel, B. DeVreese, C. Petrisor, and P. Demeester, "On the Influence of Best-Effort Network Conditions on the Perceived Speech Quality of VoIP Connections," Proceedings of IEEE 10th International Conference of Computer Communications and Networks, Scottsdale, AZ, October 2001, pp. 334-339.

[7] W. Jiang, K. Koguchi, and H. Schulzrinne, "QoS Evaluation of VoIP End-Points," Proceedings of IEEE International Conference on Communications, ICC'03, Anchorage, May 2003, pp. 1917-1921

[8] Avaya Inc., "Avaya IP voice quality network requirements," http://www1.avaya.com/enterprise/whitepapers, 2001.

[9] A. Markopoulou, F. Tobagi, M. Karam, "Assessing the quality of voice communications over internet backbones", *IEEE/ACM Transaction on Networking*, vol. 11, no. 5, 2003, pp. 747-760

[10] Recommendation G.711, "Pulse Code Modulation (PCM) of Voice Frequencies," ITU, November 1988

[11] Recommendation H.323, "Packet-based Multimedia Communication Systems," ITU, 1997.

[12] Recommendation G.114, "One-Way Transmission Time," ITU, 1996.

[13] ITU-T Recommendation P.800, "Methods for subjective determination of transmission quality," www.itu.in/publications/main_publ/itut.html

[14] L. Sun and E. C.. Ifeachor, "Prediction of Perceived Conversational Speech Quatlity and Effects of Playout Buffer Algorithms," Proceedings of International Conference on Communications, ICC'03, Anchorage, May 2003, pp. 1-6

[15] A. Takahasi, H. Yoshino, and N. Kitawaki, "Perceptual QoS Assessment Technologies for VoIP," *IEEE Communications Magazine*, vol. 42, no. 7, July 2004, pp. 28-34

[16] J. Walker and J. Hicks, "Planning for VoIP," NetIQ Corporation white paper, December 2002, http://www.telnetnetworks.ca/products/netIq/whitepapers/planning_for_voip.pdf

[17] Recommendation G.726, "40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)," ITU, December 1990.

[18] Recommendation G.723.1, "Speech Coders: Dual Rate Speech Coder for Multimedia Communication Transmitting at 5.3 and 6.3 kbit/s," ITU, March 1996.

[19] Annex to Recommendation G.729, "Coding of Speech at 8kbit/s using Conjugate Structure Algebraic-Code-Excited Linear-Prediction (CS-ACELP)," Annex A: "Reduced Complexity 8 kbit/s CS-ACELP Speech Codec", ITU, November 1996.

[20] W. Jiang and H. Schulzrinne, "Comparison and Optimization of Packet Loss Repair Methods on VoIP Perceived Quality under Bursty Loss," Proceedings of ACM 12th International Workshop on Network and Operating System Support for Digital Audio and Video, Miami, FL, May 2002, pp. 73-81

[21] J. S. Han, S. J. Ahn, and J. W. Chung, "Study of Delay Patterns of Weighted Voice Traffic of End-to-End Users on the VoIP Network," *International Journal of Network Management*, vol. 12, no. 5, May 2002, pp. 271-280 (2002)

[22] J. H. James, B. Chen, and L. Garrison, "Implementing VoIP: A Voice Transmission Performance Progress Report," *IEEE Communications Magazine*, vol. 42, no. 7, July 2004, pp. 36-41

[23] W. Jiang and H. Schulzrinne, "Assessment of VoIP Service Availability in the Current Internet" Proceedings of International Workshop on Passive and Active Measurement (PAM2003), San Diago, CA, April 2003.

[24] M. Karam and F. Tobagi, "Analysis of delay and delay jitter of voice traffic in the Internet," *Computer Networks Magazine*, vol. 40, no. 6, December 2002, pp. 711-726 (2002)

[25] S. Riley and R. Breyer, "*Switched, Fast, and Gigabit Ethernet*," Macmillan Technical Publishing, 3rd Edition, 2000.

[26] CAIDA, http://www.caida.org/tools/taxonomy, April 2004.

[27] R. Prasad, C. Dovrolis, M. Murray, and K.C. Claffy, "Bandwidth Estimation: Metrics, Measurement Techniques, and Tools," *IEEE Network Magazine*, vol. 17, no. 6, December 2003, pp. 27-35

[28] Cisco Systems Inc., "Cisco 2621 Modular Access Router Security Policy," 2001, http://www.cisco.com/univercd/cc/td/doc/product/access/acs_mod/cis2600/secure/2621rect.pdf

[29] 3Com, "3Com Networking Product Guide," April 2004, http://www.3com.co.kr/products/pdf/productguide.pdf

[30] K. M. Chandy and C. H. Sauer, "Approximate methods for analyzing queueing network models of computing systems," *Journal of ACM Computing Surveys*, vol. 10, no. 3, September 1978, pp. 281-317.

[31] F. Gebali, *Computing Communication Networks: Analysis and Designs*, Northstar Digital Design, Inc., 3rd Edition, 2005.

[32] L. Kleinrock, Queueing Systems: Theory, vol 1, New York, Wiley, 1975.

[33] W. Leland, M. Taqqu, W. Willinger, D. Wilson, "On the Self-Similar Nature of Ethernet Traffic", *IEEE/ACM Transaction on Networking*, vol. 2, no. 1, February 1994, pp. 1-15

[34] R. Suri, "Robustness of Queueing Network Formulas," *Journal of the ACM*, vol. 30, no. 3, July 1983, pp. 564-594.

[35] R. Onvural, "Survey of Closed Queueing Networks with Blocking," *ACM Computing Surveys*, vol. 22, no. 2, June 1990, pp. 83-121

[36] J. Bolot, "End-to-End Packet Delay and Loss Behavior in the Internet," Proceedings of ACM Conference on Communications, Architectures, Protocols and Applications, San Francisco, CA, October 1993, pp. 289-298

[37] OPNET Technologies, http://www.mil3.com

[38] K. Pawlikowski, H. Jeong and J. Lee, "On Credibility of Simulation Studies of Telecommunication Networks", *IEEE Communications Magazine*, vol. 40, no. 1, January 2002, pp. 132-139

[39] A. Law and W. Kelton, *Simulation Modeling and Analysis*, McGraw-Hill, 2nd Edition, 1991.