An Analytic Approach for Deploying Desktop Videoconferencing

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

Abstract. The deployment of desktop videoconferencing, also known as Video and Voice over IP (VVoIP), over existing IP networks is gaining popularity these days. Such a deployment has become a major and challenging task for data network researchers and designers. This paper presents an analytic approach for deploying videoconferencing. The approach utilizes queueing network analysis and investigates two key performance bounds for videoconferencing: delay and bandwidth. The approach can be used to assess the support and readiness of an existing IP network. Prior to the purchase and deployment of desktop videoconferencing equipment, the approach predicts the number of videoconferencing sessions or 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 our approach to a typical network of a small enterprise. In addition, we use OPNET network simulator to verify and validate our analysis. Results obtained from analysis and simulation are in line and give a close match.

Keywords: VoIP, VVoIP, Videoconferencing, Network Design, Performance Evaluation, Queueing Analysis

1 Introduction

Desktop videoconferencing or video telephony, also known as Video and Voice over IP (VVoIP), is a vital tool that provides natural, effective and powerful communication. The deployment of desktop videoconferencing over existing IP networks in both industry and academia has been increasing rapidly. Desktop videoconferencing applications range from internal company communications, educating and training remote employees, to telecommuting. It can eliminate certain travel requirements, thereby cutting costs. Desktop videoconferencing takes advantage of a key workplace tool, that is the PC. In the past few years, an H.323 standard was introduced by the ITU, and thus paved the way to the fast growth and deployment of videoconferencing. H.323 is a full suite of protocols developed by ITU to define how real-time multimedia communications, such as videoconferencing, can be exchanged over data or packet-switched networks [1].

Many network managers are finding it very advantageous and cost effective to deploy desktop videoconferencing over their existing IP networks. 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.

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

On the other hand, videoconferencing requires timely packet delivery with low latency, jitter, packet loss, and sufficient bandwidth. To achieve this goal, an efficient deployment of videoconferencing must ensure these real-time traffic requirements can be guaranteed over new or existing IP networks.

Because videoconferencing places a high demand on network resources, many network architects, managers, planners, designers, and engineers are faced with common strategic, and sometimes challenging, questions. What are the QoS requirements for videoconferencing? How will the new videoconferencing load impact the QoS for currently running network services and applications? Will my existing network support videoconferencing and satisfy the standardized QoS requirements? If so, how many videoconferencing sessions can the network support before upgrading prematurely any part of the existing network hardware?

Some commercial tools were developed for testing the performance of multimedia applications in data networks. A list of the available commercial tools that support VVoIP is listed in [2-6]. For the most part, these tools use two common approaches in assessing the deployment of VVoIP into the existing network. One approach is based on first performing network measurements and then predicting the network readiness for supporting VVoIP. The prediction of the network readiness is based on assessing the health of network elements. The second approach is based on injecting real voice and video traffic into existing network and measuring the resulting delay, jitter, and loss.

None of these commercial tools offer an analytic approach for successful videoconferencing deployment. In particular, none gives any prediction for the total number of videoconferencing calls that can be supported by the network taking into account important design and engineering factors such as flow of calls and their distribution, future growth capacity, performance thresholds, and impact of adding videoconferencing on existing network services and applications. This paper attempts to address those important factors using an analytic approach based on queueing networks. The paper contains many useful engineering and design guidelines, and discusses many practical issues pertaining to the deployment of videoconferencing. These issues include characteristics of VVoIP traffic and QoS requirements, 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.

In previously related work [7], a methodology was presented for the deployment of Voice over IP (VoIP) in existing Ethernet networks. In sharp contrast to this previous work, this paper is different in significant ways. First this paper focuses on discussing an analytic approach and not a comprehensive methodology. Second, the analytic approach is for the deployment of desktop videoconferences and not for VoIP. Third, the characteristics and requirements (such as bandwidth and delays) of deploying videoconferencing are quite different than that of VoIP. The paper discusses those issues and shows how different they are from VoIP. Third, the simulation configuration, setup, and generation of traffic for videoconferencing are considerably different than that of VoIP when considering the deployment of both voice and video calls simultaneously. Fourth, the call or session distribution is different. In this paper, we consider only *intranet* videoconferencing sessions. For comparison purposes, we will apply our analysis to the same Ethernet network topology discussed in [7]. We will also use the same background traffic.

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 desktop videoconferencing. Section 3 describes key issues and requirements that have to be defined and tackled upfront. Section 4 presents our analytic approach for deploying successfully videoconferencing. Section 5 describes and summarizes the OPNET model and results. Section 6 describes important design and engineering decisions to be made based on the analytic study. Section 7 concludes the study and identifies future work.

2 Existing Network

A topology of typical network of a small- to medium-sized company residing in a high-rise building is shown in Figure 1. 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.

3 Key Issues and Requirements

Deployment of desktop videoconferences requires a sound understanding of a number of issues. These issues need to be identified and tackled early on. These are design issues and considerations that feed into our analysis and simulation to determine the actual number of videoconferencing sessions that can be supported by an IP network.

3.1 Traffic Characteristics and Requirements

For introducing a new network service such as desktop videoconferencing, one has to characterize first the nature of its traffic, QoS requirements, and any additional components or devices. In this paper, we assume a point-to-point desktop videoconferencing. Streaming stored video and broadcast video [8] is not considered in this paper and is left for future work. An H.323 *gatekeeper* or *CallManager* node, which is an optional component, is typically added to the network [1,9,10]. The *gatekeeper* node handles signaling for establishing, terminating, and authorizing connections of video sessions, as well as imposing maximum bandwidth for each session. Other hardware requirements include an H.323 workstation or multimedia PCs. A multimedia PC has H.323 voice and video software and is equipped with a camera and a microphone.

3.1.1 Bandwidth

Point-to-point IP videoconference session consists of two independent bidirectional streams: voice and video [11]. The required bandwidth for a voice call, one direction, is 50 pps (packets per second) or 90.4 kbps (bits per second). 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.

As opposed to the fixed packet size of voice, the packet size for video is variable as the video data are highly correlated. The nature of H.323 video is to send images to receiver, then subsequently send only update to the images, and thus conserving considerable bandwidth. Typically, for a single video call, the required bandwidth is 30 frames/s or fps. In Ethernet, one video frame is packetized in one Ethernet frame with sizes ranging from 65-1518 bytes. According to real measurements performed by [9,12], the most common size is ranging from 1025-1518 bytes. To be conservative, we will assume a video frame size of 1344 with a rate of 30 fps for our analytic and simulation work. This gives approximately a rate of 320 kbps for pure video traffic. A bandwidth of 320kbps is a multiple of the basic 64 kbps communication channel and is an acceptable bandwidth for business-quality desktop videoconferencing with default recommendations of H.261 video codec, CIF video resolution, and H.323 frame rate of 30 fps [8,10,13,14]. When considering the additional 66 bytes of layer headers, as specified in [15] and happens to be similar to byte overhead for VoIP, the required bandwidth for a video call would be 338.4 kbps. For both directions, the required bandwidth for a single video call is 60 pps or 676.8 kbps assuming a symmetric flow. Hence, for a bidirectional videoconferencing session the required bandwidth is 160 pps or 857.6 kbps.

3.1.2 End-to-End Delay

In order to achieve a natural interactive videoconferencing session, the end-to-end upper bound delay (sometimes termed latency) for a video or voice packet should be kept to minimal. Essentially, such a delay can broken into at least three contributing components, which are as follows (i) voice sampling or frame grabbing, 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.

According to recommendations by ITU [16], when delays are less than 150 ms, most interactive applications, both speech and non-speech, will experience essentially transparent interactivity. For voice, the end-to-end delay is sometimes referred to by M2E or Mouth-to-Ear delay [17]. In videoconferencing, there is no separate delay for voice and video streams as both voice and video are synchronized in what is commonly known as "lip-sync". Therefore, for our upper bound end-to-end one-way delay of a video or voice packet, we will use 150 ms.

3.1.3 Additional Considerations

Throughout our analysis and work, we assume voice and video calls are symmetric. We also ignore the signaling traffic generated by the *gatekeeper*. We base our analysis and design on the worst-case scenario for videoconferencing traffic. The signaling traffic involving the *gatekeeper* is only generated prior to the establishment of the session and when the session is finished. This traffic is relatively limited and small compared to the actual voice call traffic. In general, the *gatekeeper* generates no signaling traffic throughout the duration of the videoconferencing session for an already established on-going session [18]. In order to allow for future growth, we will consider a 25% growth factor for all network elements including router, switches, and links. This factor will be taken into account in our analysis and simulation study.

3.2 Traffic Flow and Session Distribution

Specifying the flow of sessions and their distribution is an important step that plays a factor in determining the number of sessions to be supported. Traffic flow has to do with the path that a session travels through. Session distribution has to do with the percentage of sessions to be established within and outside of a floor, building, or department. For our example, we will assume that the generation of sessions is symmetric for all three floors. The intra-floor traffic will constitute 20% of over all traffic, and the other 80% will constitute inter-floor traffic. Such a distribution can be described in a simple probability tree shown in Figure 2. Some important observations can be made about the voice traffic flow for inter-floor sessions. For all these type of calls, the 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 sessions 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.

3.3 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 analytic study and simulation. 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 and pps 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 1. Worst-case network measurements

Link	Mbps	pps
Router ⇔ Switch 1	9.44	812
Router ⇔ Switch 2	9.99	869
Switch 1 ⇔ Floor 1	3.05	283
Switch 1 ⇔ Floor 2	3.19	268
Switch 1 ⇔ File Server	1.89	153
Switch 1 ⇔ DB Server	2.19	172
Switch 2 ⇔ Floor 3	3.73	312
Switch 2 ⇔ Email Server	2.12	191
Switch 2 ⇔ HTTP Server	1.86	161
Switch 2 ⇔ Firewall	2.11	180
Switch 2 ⇔ Proxy	1.97	176

3.4 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 Section 3.3, all the links connecting the router and the switches and links connecting the servers and the switches are underutilized. If any of the links was heavily utilized, e.g. 30-50%, the network engineer should decide to re-dimension the link to 1-Gbps link at this stage. Shared links of Floor 1, Floor 2, and Floor 3 must be replaced. Real-time applications such as voice and video should never be deployed on networks that have shared Ethernet links. Shared Ethernet scales poorly and offers zero QoS [19]. In order to consistently maintain the required QoS, a switched fast full-duplex Ethernet LAN becomes necessary.

Based on the hardware requirement for deploying videoconferencing described in Section 3.1, an H.323 *gatekeeper* node has to be added to the existing network. As a network design issue, an appropriate node placement is required for this node. 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. It is also 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 videoconferencing sessions. Figure 3 shows the new network topology with the addition of the *gatekeeper* and the replacement of three shared Ethernet LANs with 100Mbps switched Ethernet LANs.

4 Analytic Approach

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

4.1.1 Bandwidth Bottleneck Analysis

This step identifies the network element (whether it is a node or a link) that puts a limit on how many videoconferencing calls or sessions 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 [20], this minimum available bandwidth is defined as follows

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

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.

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

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

where $growth_i$ is the growth factor of network element E_i, and takes a value from 0 to 1. $CallBW$ is the VVoIP bandwidth for a one session imposed on E_i. As previously discussed in Section 3.1.1, the bandwidth for one videoconferencing session in one direction is 80 pps or 428.8 kbps. In order to find the bottleneck network element that limits the total number of VVoIP 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 VVoIP traffic flow passing by this element. The percentage of traffic load imposed on E_i, denoted as $load_i$, can be found by examining the distribution of the calls. The total number of VVoIP calls that can be supported by a network can be expressed as

$$TotalCallsSupported = \min_{i=1,\dots,N}\left(\frac{MaxCalls_i}{load_i}\right). \qquad (2)$$

Let us for the sake of illustration compute the $MaxCalls_i$ and $load_i$ supported by the Router, Switch 1, and uplink from Switch 2 to the Router. Table 2 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 1. C_i, for the router and the switch is usually given by the product datasheets. According to [21] and [22], the capacity C_i for the router or the switch, is 25,000pps and 1.3M pps, respectively. $load_i$ is computed by examining the probability tree for call distribution shown in Figure 2.

Table 2. Maximum VVoIP calls support for few network elements

Network Element	C_i	u_i	$CallBW$	$load_i$	$MaxCalls_i$
Router	25,000 pps	6.72%	160 pps	4/5	109
Switch 1	1.3 Mpps	0.13%	160 pps	4/5	6,085
Switch 1 → Router uplink	100 Mbps	9.44%	428.8 kbps	4/5	158
Switch 2 → Router uplink	100 Mbps	9.99%	428.8 kbps	8/15	157

Table 2 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), $load_i$ 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 135 sessions.

For the sake of illustration, we show how u_i and $load_i$ can be computed. u_i can be computed from Table 1. 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 1, this yields to (812+869)/25000=6.72%. $load_i$ can be computed using the probability tree shown in Figure 2 as follows. For the router, $load_i$ is the percentage of the inter-floor, which is 4/5. Similarly, $load_i$ for Switch 1 and the uplink from Switch 1 to the router would be 4/5, as all inter-floor calls have to pass through them. As for the uplink form Switch 2 to the router, $load_i$ can be expressed as (4/5){1/3+1/3}.

4.1.2 Delay Analysis

As discussed earlier, the maximum tolerable end-to-end delay for a videoconferencing packet is 150 ms (see Section 3.1.2). The maximum number of videoconferencing sessions 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 deploying VVoIP, i.e. the maximum number of videoconferencing sessions that an existing network can support while maintaining VVoIP QoS. This can be done by adding calls incrementally to the network while monitoring the threshold or bound for VVoIP delay. When the end-to-end delay, including network delay, becomes larger than 150 ms, the maximum number of videoconferencing sessions can then be known.

As described in Section 3.1.2, there are three sources of delay for a VVoIP stream: sender, network, and receiver. An equation is given in [23] 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 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 VVoIP 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 [24]. 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 VVoIP traffic to be Poisson. In reality, the inter-arrival time, with mean of $1/\lambda$, of voice stream is constant, and for the video stream is fluctuating as it has variable video frame. However, assuming a Poisson voice arrival gives adequate approximation according to [23], especially when employing a high number of calls. For simplification purposes, we also assume Poisson arrival for video. It is to be noted that 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 4 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 [21,22], 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. [25] provides a comprehensive models of common types of switches and routers. According to [26], the average delay for a 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. As discussed earlier, the queueing models in Figure 4 assume Poisson arrival for both voice, video and background traffic.

It is worth noting that the analysis by decomposition of queueing networks in [24] assumes exponential service times for all network elements including links. But [27] 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. [27] 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 to the incoming link of receiver PC. 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 5 shows an example of how to compute the network delay. Figure 5(a) shows the path of a unidirectional voice traffic flow going from Floor 1 to Floor 3. Figure 5(b) shows the corresponding networking queueing model for such a path.

For Figure 5(b), in order to compute the end-to-end delay for a single bi-directional videoconferencing 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. $\lambda=\lambda_{voice}+\lambda_{video}+\lambda_{bg}$, where λ_{voice} is the total added new traffic of a single voice call in pps, λ_{video} is the total added new traffic of a single video call in pps, and λ_{bg} is the background traffic in pps. For an uplink or downlink, $\mu=(1-25\%)\times 100\text{Mbps}$, $\lambda=\lambda_{voice}+\lambda_{video}+\lambda_{bg}$. Since the service rate is in bps, λ_{voice}, λ_{video}, and λ_{bg} must be expressed in bps. From Table 1, one can express the bandwidth for background traffic and for a single call in both pps and bps. Similarly for the router, $\mu=(1-25\%)\times 25{,}000\text{pps}$ and $\lambda=\lambda_{voice}+\lambda_{video}+\lambda_{bg}$. Here λ_{voice}, λ_{video}, and λ_{bg} must be expressed in pps. As discussed in Section 3.1.1, for a single bi-directional videoconferencing call, λ_{voice} at the router and switches for a single call will be equal to 100pps. For λ_{video}, it is equal to 60pps. However, for the uplink and downlink links, it is 90.4 kbps for λ_{voice} and 676.8 kbps for λ_{video}. For multimedia PCs, we assume each PC introduces a λ_{bg} of 10% of the total background traffic utilized by the floor that the PC is located in.

The total delay for a single VVoIP call of Figure 5(b), 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 videoconferencing sessions that can be supported by an existing network while maintaining videoconferencing delay constraint, we developed the following algorithm that basically determines network capacity in terms of VVoIP 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 session (a voice call and a video call) is added, according to the call distribution shown in Figure 2.
iii) For each network element, $\lambda = \lambda_{voice} + \lambda_{video} + \lambda_{bg}$ is computed. λ_{bg} is known for each element; however, λ_{voice} and λ_{video} 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 VVoIP packet is computed.
v) The end-to-end delay is computed by summing up all the delays of step (iv) encountered for each possible VVoIP flow. This includes all 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 videoconferencing sessions 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 6. It can be observed from the figure that the delay increases sharply when the number of calls go beyond 134 calls. To be more precise, MATLAB results showed the number of calls that are bounded by the 80 ms delay is 136.

When comparing the number of sessions that network can sustain based on bottleneck bandwidth and worst-delay analysis, we find the number of sessions is limited by the available bandwidth more than the delay, though the difference is small. Therefore, we can conclude that the maximum number of sessions that can be sustained by the existing network under study is 135.

Packet Loss. A question related to determining the number of calls to be supported by a particular data network is packet loss. VVoIP packet loss should be below 1% according to [28], 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 [29]. Determining packet loss for this type of networks is not a trivial task, and can be only approximated based on [29,30]. 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 [29]. Due to lack of closed-form analytical solutions and according to [30], 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.

5 Simulation

The object of the simulation is to verify analysis results of supporting VVoIP calls. We used the popular MIL3's OPNET Modeler simulation package[1], Release 8.0.C [31]. 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.

5.1 Modeling the Network

A snapshot of the OPNET simulation model for the existing network under study is shown in Figure 7. 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. The enterprise servers are modeled as Ethernet servers. All network elements have been connected using a 100 Base-T links. Figure 7 (main plot) 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. Floor LANs have been modeled as subnets that enclose an Ethernet switch and three designated Ethernet workstations used to model the activities of the LAN users, as shown in Figure 7 (upper-left-corner frame). For example, the Ethernet workstations for Floor 1 are labeled as `F1_C2`, `F1_C2`, and `F1_C3`. `F1_C1` is a source for sending voice and video calls. `F1_C2` is a sink for receiving voice and video calls. `F1_C3` is a sink and source of background traffic. The voice and video traffic were generated according to the flow and call distribution discussed in Section 3.2. Other 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 voice and video traffic. For traffic generation, two applications and a profile have to be created. OPNET has two built-in or predefined applications for voice and video, that is `VoIP_APPLICATION` and `VC_APPLICATION`, respectively, as shown in Figure 7.

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

5.2 Simulation Results

OPNET has to be configured to obtain graphed results for numerous network components which include voice and video traffic, router, switches, and links. In this section, graphed results for some of the most important components are shown. We configured the duration of the OPNET simulation run for 4 minutes. Per our configurations, the generation of background traffic, by default in OPNET, started at 40 seconds from the start time of the simulation run. The videoconferencing traffic of both voice and video started at 70 seconds at which a total of 6 calls are initially added (3 for voice and 3 for video). Then, every 2 seconds another 6 calls are added (see Figure 8 and 9). Figure 8 verifies that at 70 seconds of simulation time 3 voice calls and 3 video calls are added, and thus producing traffic load of 300 packets for voice and another 180 packets for video. This is expected and is in line with analysis since the bandwidth for a single bidirectional.voice call is 100 pps and for a single bidirectional video call is 60 pps.

The Simulation stops at 4 minutes, reaching maximum calls at actually 3 minutes and 58 seconds. At 3.58 minutes, the maximum number of calls is $6+((3\times 60+58-70)/2)\times 6=510$. Ideally, this would translate into 255 calls for voice or (255x100=25,500) voice packets, and another 255 calls for video or (255x60=15,300) video packets or frames. It is clear from Figure 9 that the maximum packet rate for voice and video that were generated at the end of simulation run was close to these numbers. Figure 9 shows the behavior of global videoconferencing traffic as voice and video calls are added every two seconds to the network by the designated workstations. The figure shows the traffic in pps that was sent, received, and dropped.

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 9 shows clearly that not all of packets being sent get received, as there is a mismatch between traffic sent and received. We can determine the number of calls that can be supported by examining the X and Y axes. When zooming in and examining the X axis of the simulation run time, it is clear that the last successful addition of three calls was at exactly 2 minutes and 56 seconds. The next addition, as shown, was at 2 minutes and 58 seconds and resulted in a mismatch. This mismatch point happens to be true for both voice and video traffic. One can determine the number of calls to be supported by the network for voice and video simply by calculating how many calls have been added until the last successful addition of three calls, i.e. at 2 minutes and 56 seconds. Since the last successful addition point was the same for voice and video, this yields to $3+((2\times 60+56-70)/2)\times 3=162$ videoconferencing calls.

Figure 10 shows the corresponding voice and video end-to-end delay of Figure 9. Here the end-to-end delay is essentially the network delay, as the OPNET model does not count for jitter and processing delay at the endpoints. It is to be noted that the voice and video traffic for each floor was generated by its designated workstation which was configured with infinite CPU power and memory. For Figures 10((a) and (b)), the delay is reported as 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. As discussed in Section 4.1.2, the end-to-end network delay should not exceed 80 ms for voice or video packets. The delay for voice and video, as shown in Figure 10, behave similarly for the most part because both packets follow the same paths. When zooming in on the area of 3 minutes of Figure 10, it is depicted that the delay stays less than 80 ms until a simulation time of 3 minutes and 2 seconds at which the delay increases sharply. One can then determine the number of calls that the network can support to satisfy the 80 ms time constraint. The number of videoconferencing calls can be computed as $3+((3\times 60+2-70)/2)\times 3=171$. Therefore one can conclude that, based on these simulation results, the number of videoconferencing calls to be supported by the network is bounded more by the network bandwidth than the delay. Hence, the number of videoconferencing sessions that the network can support based on simulation is 162 calls.

Based on the simulation results, the existing network can support 162 videoconferencing calls while satisfying the videoconferencing QoS of throughput, latency and packet loss. In Section 3.7.2, videoconferencing calls were added every 2 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 162 videoconferencing 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 simulation run of supporting 162 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 144 calls showed compliant and healthy results with no packet loss, end-to-end delay of 10.50 ms, and adequate utilization of router and switch CPUs and links.

6 Engineering Decisions

For the IP network under study, the following network design and engineering decisions can be justified:

- Based on analytic approach a total of 135 videoconferencing sessions can be supported.
- Based on simulation a total of 144 videoconferencing sessions can be supported.
- Consequently and to be on the conservative side, the network, with a reserved growth factor of 25%, can *safely* support up to 135 video while meeting the videoconferencing QoS requirements and having no negative impact on the performance of existing network services or applications.
- For 135 calls, a network delay of about 10 ms is encountered. To be precise, analysis gave a delay of 9.84 ms, while simulation gave a delay of 10.50 ms.
- 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 135 calls, the router replacement is a must. The router can be replaced with a popular Layer-3 Ethernet switch. Before prematurely changing other network components, one has to find out how many videoconferencing sessions 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.

7 Concluding Remarks

The paper presented an analytic approach to assess network readiness and support for deploying desktop videoconferencing. The approach can help network researchers and designers to determine quickly and easily how well VVoIP will perform on a network prior to deployment. Prior to the purchase and deployment of VVoIP equipment, it is possible to predict the number of VVoIP calls (or videoconferencing sessions) 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 desktop videoconferencing. These issues include characteristics of VVoIP traffic and QoS requirements, VVoIP flow and call distribution, defining future growth capacity, and measurement and impact of background traffic.

As a case study, we applied our approach to a typical network of a small enterprise. In addition we used OPNET network simulator to verify and validate our analysis. Results obtained from analysis and simulation were in line. There was only a difference of 9 sessions between analysis and simulation. 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 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.

Only point-to-point videoconferencing was considered in this paper. As a future work, one can consider deploying broadcast and multicast videoconferencing. Also as a future work, one can look into assessing the network support and readiness of deploying other popular real-time network services such as streaming stored video and web conferencing. As a near-term work, we are in the process of developing a GUI-based design tool that automates the analytic approach presented in this paper in order to find the maximum number of VVoIP calls that can be supported by a given generic network.

Acknowledgement

The author acknowledges the support of King Fahd University of Petroleum and Minerals in the development of this work. The author also acknowledges the anonymous reviewers for their valuable comments on the earlier versions of this article.

References

[1] Recommendation H.323, “Packet-based Multimedia Communication Systems,” ITU, 1997.

[2] 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

[3] M. Bearden, “Assessing Enterprise Network Readiness for Video-and-Voice-Over-IP”, White Paper, Applied Global Technologies, Inc., May 2004.

[4] 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.

[5] J. Klaue, B. Rathke, and A. Wolisz, "EvalVid - A Framework for Video Transmission and Quality Evaluation", In Proc. of 13th International Conference on Modeling Techniques and Tools for Computer Performance Evaluation, September 2003.

[6] Brix Networks - http://www.brixnetworks.com

[7] K. Salah, “On the deployment of VoIP in Ethernet Networks: Methodology and Case Study,” *Computer Communications Journal*, Elsevier Publication, Accepted.

[8] B. Liao, “IP Videoconferencing Embracing the Next Era of Visual Collaborations,” Proceedings of the 5th IEEE Asia-Pacific Conference on Optoelectronics and Communications Conference, Beijing, October 1999, pp. 853-863

[9] Cisco Systems, “Implementing QoS Solutions for H.323 Video Conferencing over IP,” Technical Report, Document ID: 2155, available at: http://www.cisco.com/warp/public/105/video-qos.html

[10] R. Abler and G. Wells, “Supporting H.323 Video and Voce in an Enterprise Network,” Proceedings of the 1st USENIX Conference on Network Administration, Santa Clara, Clifornia, April 7-10, 1999.

[11] H. Schulzrine, S. Casner, R. Frederick and V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications,” RFC 1889, January 1996.

[12] P. Calyam and C. G. Lee, “Characterizing voice and video traffic behavior on the Internet”, Proceedings of the International Symposium on Computer and Information Sciences (ISCIS), October 2005.

[13] A. K. Alasmari, “A New Video Compression Algorithm for Differential Videoconferencing standards,” International Journal of Network Management, vol 13, no. 1, November 2003, pp. 3-10.

[14] J. Sprey, “Videoconferencing as a Communication Tool,” IEEE Transactions on Professional Communications, vol 40, no. 1, March 1997, pp. 41-47

[15] D. Tynan, “RTP Payload Format for BT.656 Video Encoding,” RFC 2431, October 1998.

[16] Recommendation G.114, “One-Way Transmission Time,” ITU, 1996.

[17] 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

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

[19] S. Riley and R. Breyer, “Switched, Fast, and Gigabit Ethernet,” Macmillan Technical Publishing, 3rd Edition, 2000.

[20] 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

[21] 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

[22] 3Com, “3Com Networking Product Guide,” April 2004, http://www.3com.co.kr/products/pdf/productguide.pdf

[23] 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)

[24] 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.

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

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

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

[28] 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

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

[30] 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

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

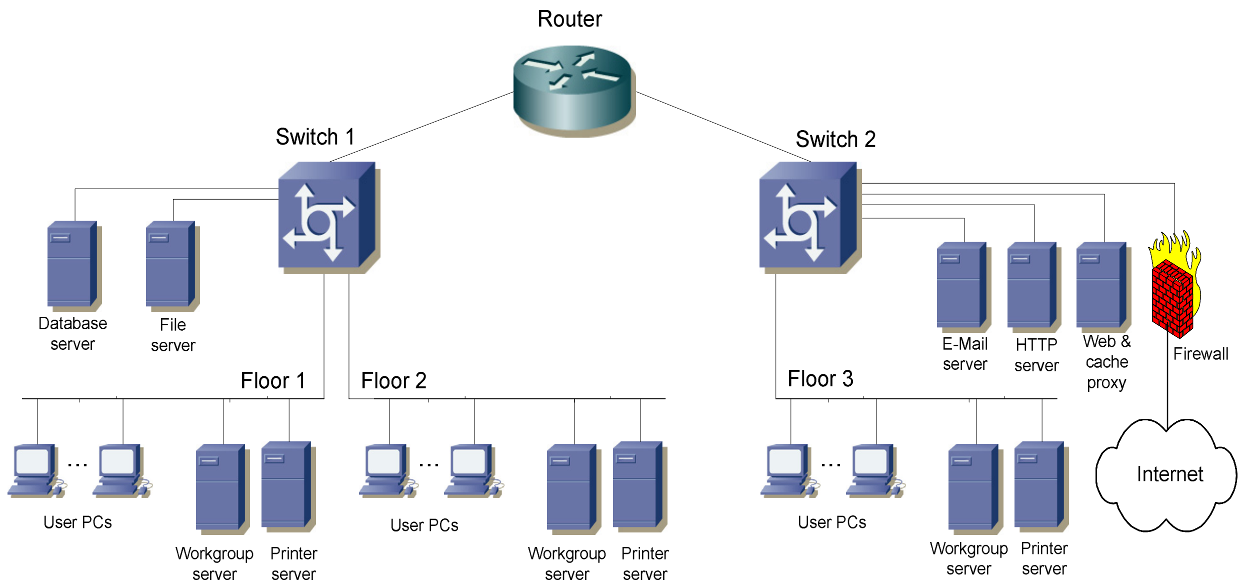


Figure 1. Logical diagram of a small enterprise

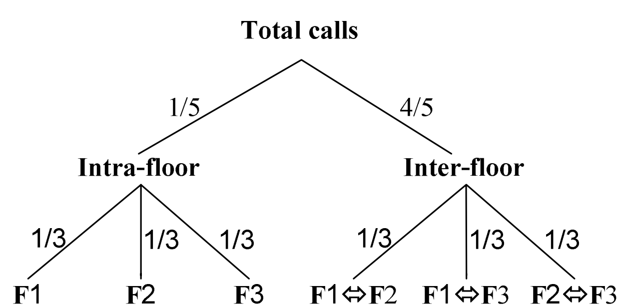


Figure 2. Probability tree describing session distribution

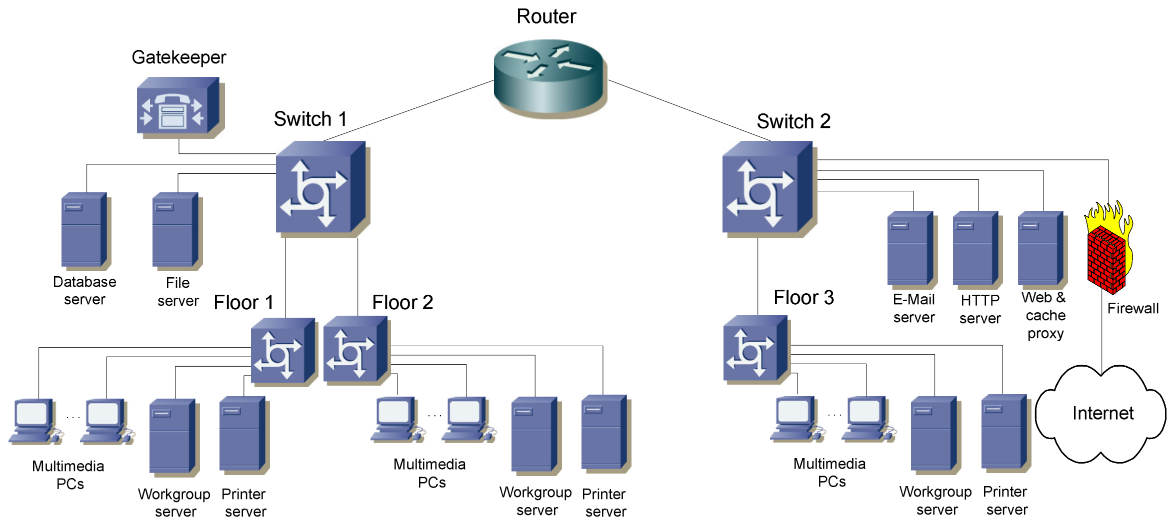


Figure 3. Network topology with added-videoconferencing Components

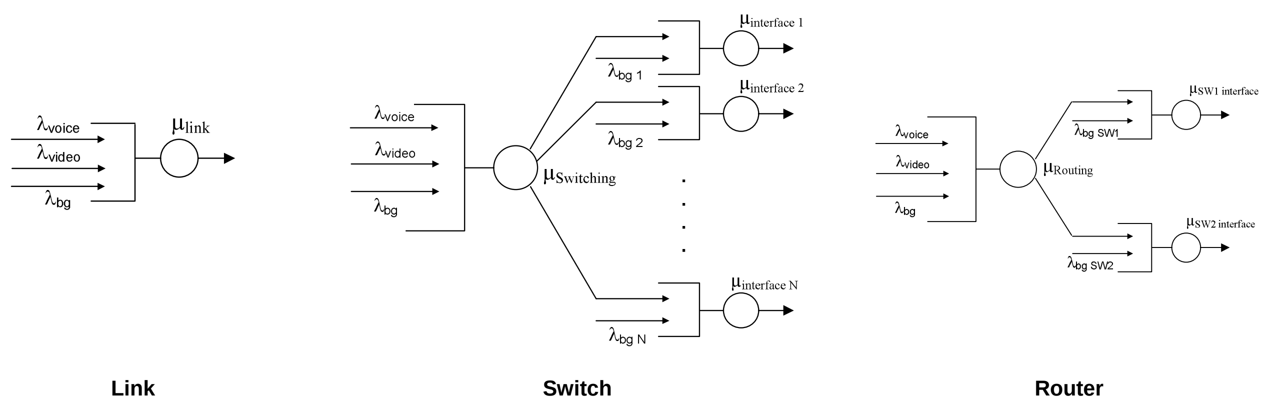
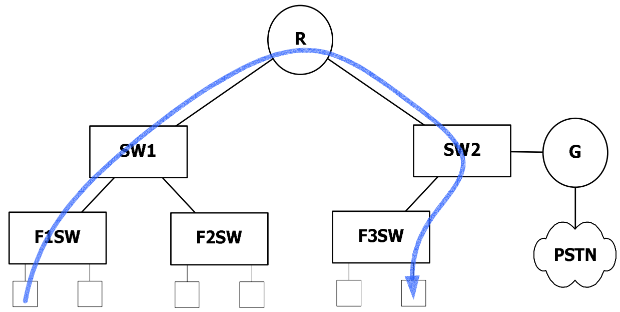
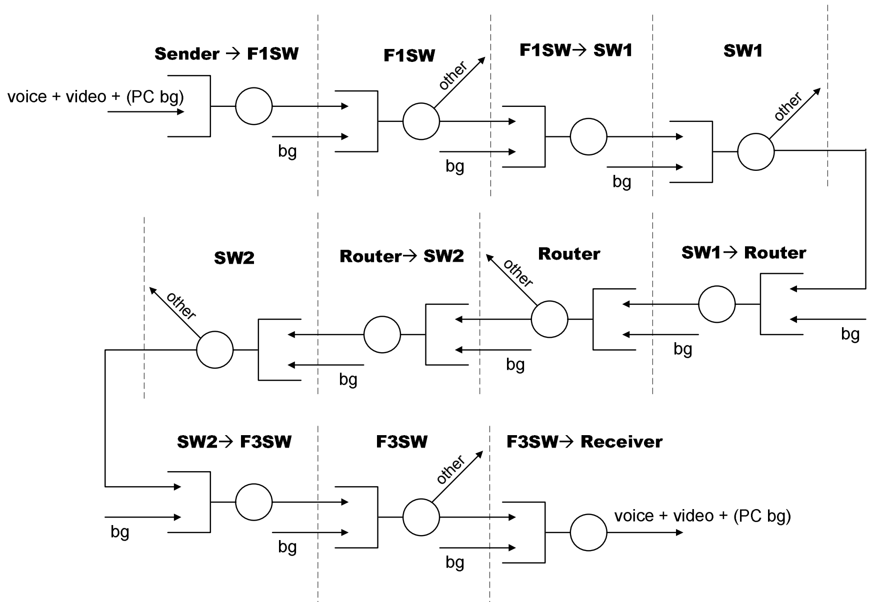


Figure 4. Queueing models for three network elements



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



(b) Corresponding network queueing model of the entire path

Figure 5. Computing network delay

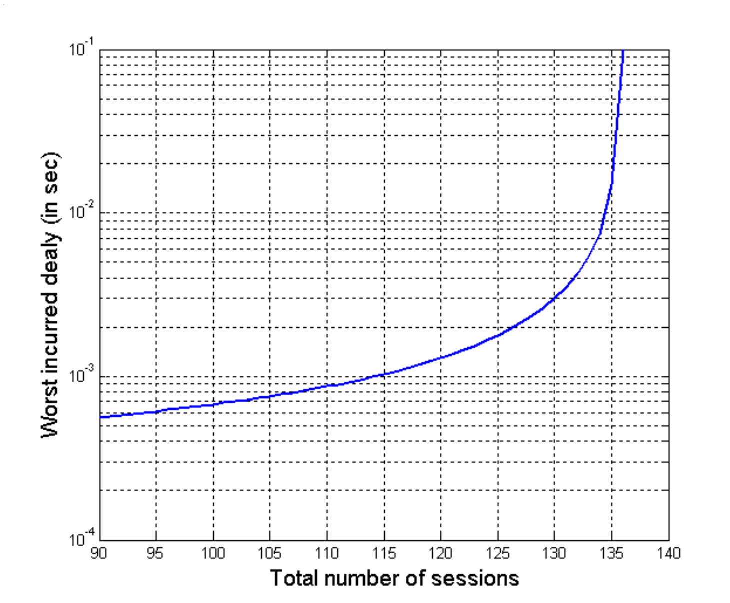


Figure 6. Worst incurred delay vs. number of sessions

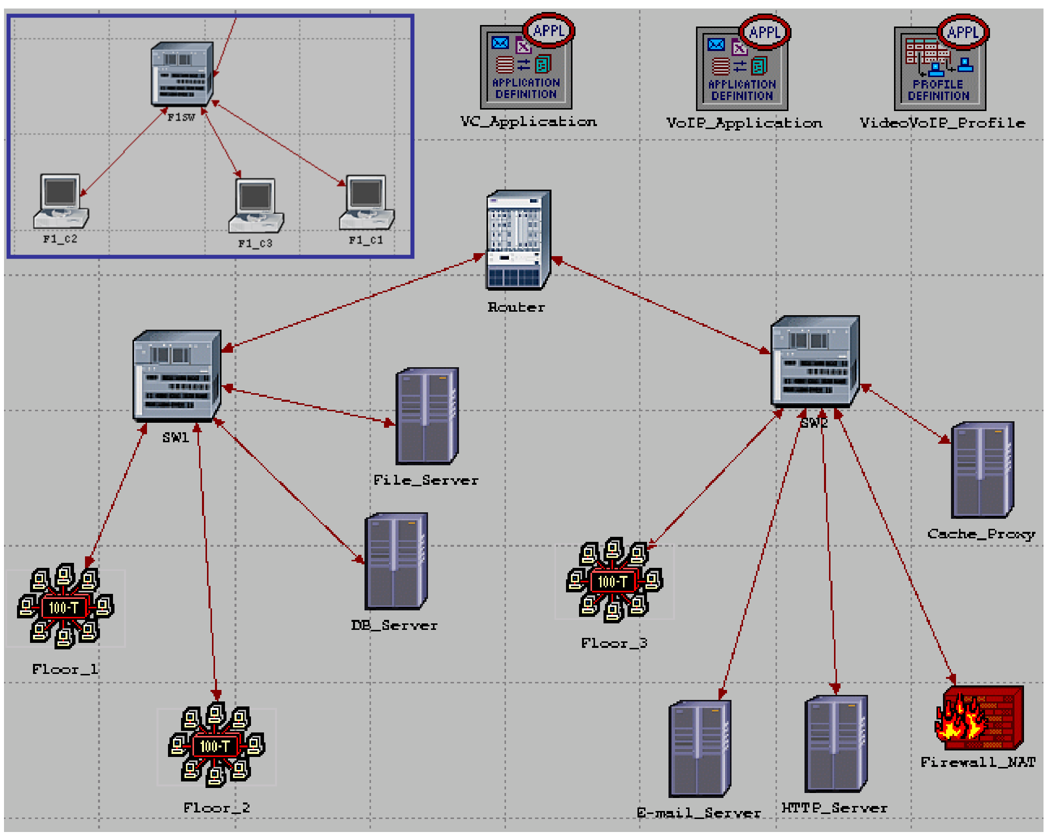


Figure 7. OPNET model of organization network with voice and video

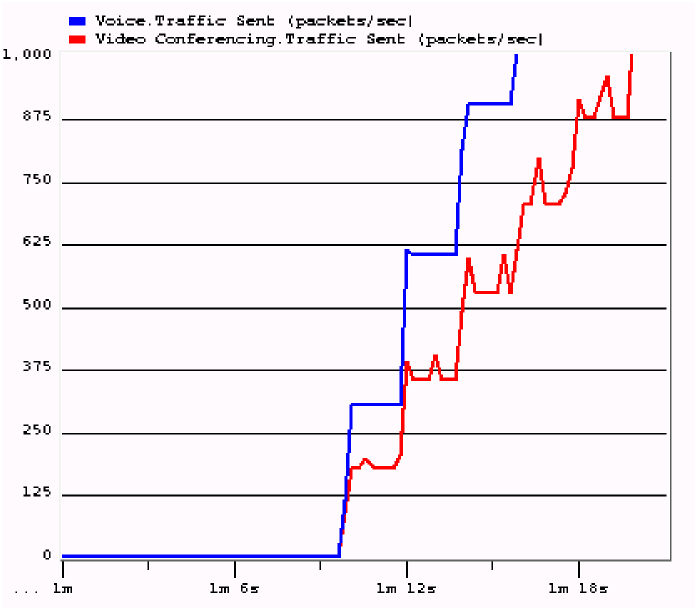


Figure 8. Generating videoconferencing load in pps

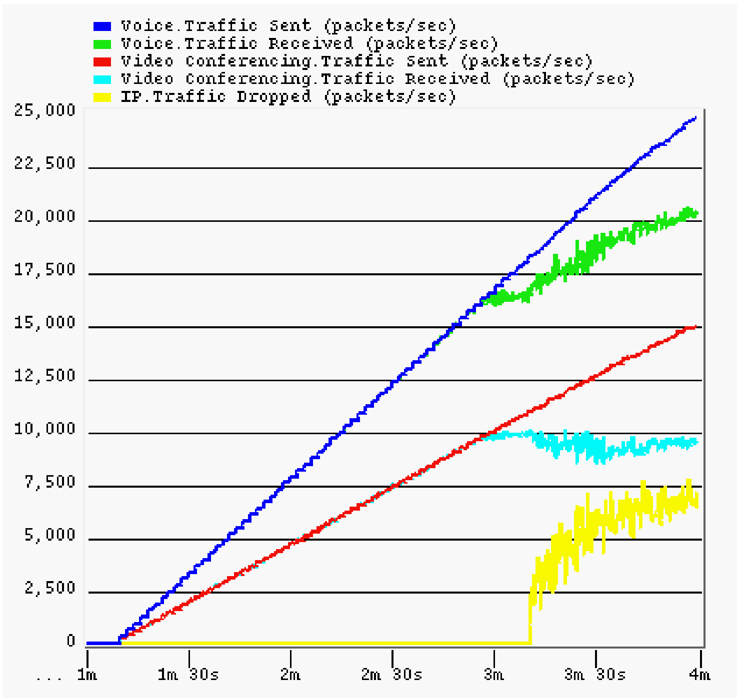
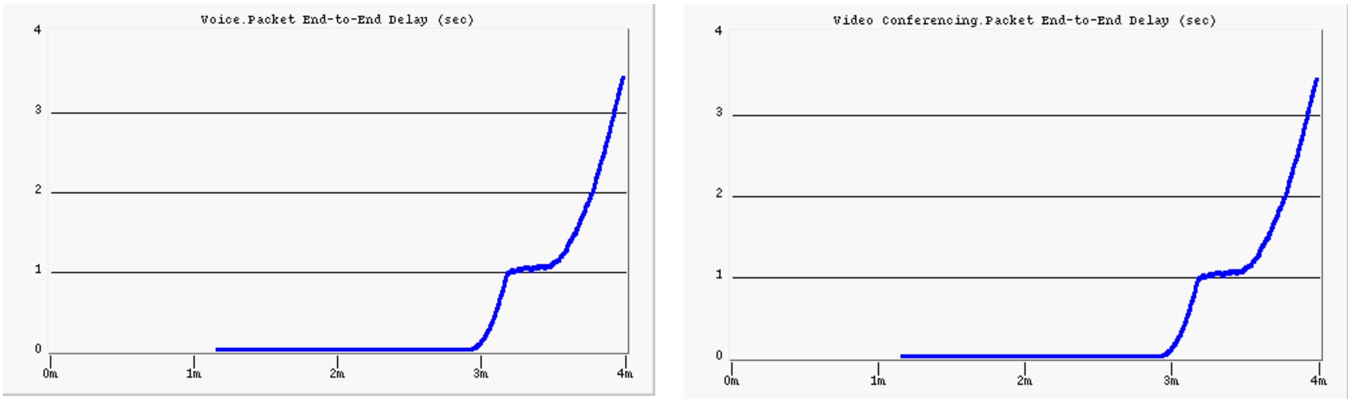


Figure 9. Global videoconferencing traffic in pps



(a) (b)

Figure 10. Global videoconferencing end-to-end delay