TRAFFIC ENGINEERING, TRAFFIC CONTROL, PERFORMANCE ANALYSIS AND NODE COMBINING IN DOCSIS-BASED CABLE NETWORKS

Gagan L. Choudhury and Moshe Segal
AT&T Labs, Middletown, NJ 07748, USA
e-mail: {gchoudhury,moshesegal}@att.com
Phone: (732)420-{3721,3713}

Abstract

Simulation and analytic models (based on Markov Chains and Linear/integer Programming) have been developed to analyze and optimally design a cable access system that carries integrated voice/data over IP and DOCSIS. The goal is a design that is cost-effective to network operators and satisfies the QoS needs of each application types.

INTRODUCTION

Over the next few years the United States and the world will see a massive deployment of cable-based access networks to carry high-speed data, voice and video services for consumers and businesses. In some cases the existing cable TV and telephony infrastructure will be upgraded, and in other cases new infrastructure will be deployed. A key to success in the marketplace would be the ability to share common resources among many subscribers (typically few tens to few hundreds) and many applications (many grades of voice, data and video) that satisfies the Quality of Service requirement of each type of subscriber and also becomes cost-effective to the service provider. In this paper we propose traffic engineering, traffic control, performance analysis and node combining methodologies to do the above. We assume that voice and data packets will be transported over the Hybrid-Fiber-Coax (HFC) architecture using IP and DOCSIS (Data over Cable Service Interface Specification) protocols. However, many of the basic observations we make in this paper are likely to be valid even if some other protocol architecture is used.

ARCHITECTURE AND DOCSIS OPERATION

The basic architecture for voice and data is shown in Figure 1. The HFC plant also carries regular television programming (not shown in the figure).

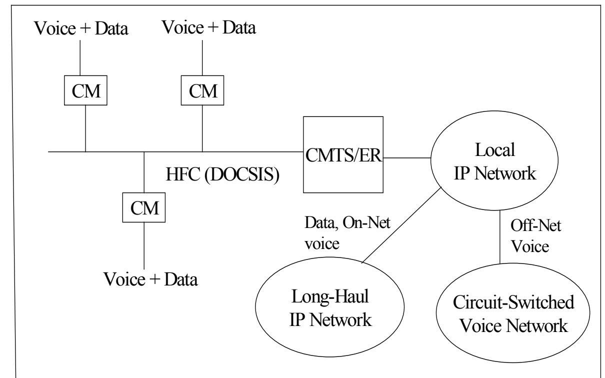

Figure 1: Basic Architecture

Each household has a cable modem (CM) which typically supports several voice lines and one data port. The voice and data are transported as IP packets (typically carrying RTP/UDP payload for voice and TCP or UDP payload for data) from the CM to the CMTS over the HFC infrastructure using the DOCSIS protocol architecture [1,2]. The first version, DOCSIS 1.0, was developed for primarily data applications. The next version, DOCSIS 1.1, has several QoS features for supporting voice and other real-time applications. The CMTS has a built-in edge-router that transports the voice and data traffic to a local IP network. From the local IP network, data typically travels to a long-haul IP network, e.g., the Internet. Voice may stay end-to-end on an IP network (On-

Net voice) or hop-off to a circuit-switched network (Off-Net voice). Usually, the HFC upstream has a frequency range 5-42 MHz (of which typically about 18 MHz is of good quality) and the HFC downstream has a frequency range 54-860 MHz. Both the upstream and the downstream frequency spectra are subdivided into many channels. Typical upstream channel widths are 1.6 MHz and 3.2 MHz even though smaller widths are also possible. Using QPSK modulation, the 1.6 MHz and the 3.2 MHz channels yield data rates of 2.56 Mbps and 5.12 Mbps respectively. Using 16QAM modulation the data rates are twice as much. The downstream channel width is 6 MHz, which yields a data rate of 30.34 Mbps using 64QAM modulation, and a data rate of 42.844 Mbps using 256QAM modulation. Upstream transmission is in bursts separated by guard times and successive bursts may be from different CMs. Each burst has to be a multiple of a basic unit known as a mini-slot that is typically 8 or 16 bytes long. Since upstream transmission is many-to-one, some coordination is needed among the transmitters. This is done by broadcasting allocation Maps periodically over the downstream with each Map precisely defining upstream transmission over a certain period of time in the future. There are three main regions of the Map as shown in Figure 2 below.

Request	Management	Data/Voice/Sig

Figure 2: Three regions of Map

The request region of the Map allows requests to be sent by all CMs for future transmission opportunities. The contention resolution among requests is similar to what is used in Ethernet [1,2]. The Management region allows management messages to be sent either by individual CMs or by all CMs in contention mode similar to the requests mentioned above. The third region allows grants for data, voice and signaling packet transmission by individual CMs (usually, a CM needs to solicit for such a grant through a successful request packet at an earlier Map). The basic QoS mechanism for tightly controlling delay jitter for voice calls is to provide periodic unsolicited grants to every active voice call. Other QoS mechanisms include real-time polling (periodic unicast request opportunities), non-real-time polling (same as real-time polling but with lower frequency) and unsolicited grants with activity detection. In order to provide QoS to a particular type of voice or data call it is necessary to define service flow IDs (SFID) on upstream and downstream channels and an associated service ID (SID) on the upstream. A SFID is a particular unidirectional mapping between a CM and the CMTS.

TRAFFIC CHARACTERISTICS

Voice Calls

Voice call arrival process is traditionally modeled as Poisson with a finite or infinite number of sources and that is appropriate in the cable environment as well. Call holding time has a general distribution including exponential, Lognormal or Pareto. The exponential case is easiest to analyze but in many cases the model output (such as blocking probability) is insensitive to the distribution except through the mean and so the other cases may be treated easily as well. In the absence of voice activity detection, voice packets are generated at a constant rate (typically once every 10 or 20 ms) throughout the call duration. In the presence of voice activity detection and silence suppression, there is a sequence of talk-spurts separated by silence periods and packets are generated (at a constant rate) only during the talk-spurts. Here again, silence and talk-spurt distributions may be general but the exponential case is easiest to analyze.

One important way in which the cable environment differs from traditional circuit-switched voice applications is that it is very natural to have multiple data rates. The payload size in a voice packet depends on the packetization interval and encoding scheme, and the overhead in the packet depends on the use or non-use of payload header suppression and the degree of forward error correction done to combat channel noise. The table below shows the data rates over the upstream under some typical scenarios. It is assumed that RTP, UDP, IP, Ethernet, CRC, MAC and physical layer overheads are 12 bytes, 8 bytes, 20 bytes, 14 bytes, 4 bytes, 11 bytes and 24 bytes respectively and under payload header suppression (PHS), all but two bytes of the UDP, IP and Ethernet layer headers are suppressed. It is also assumed that the mini-slot size is 16 bytes (note that if the payload plus overhead is not an integer multiple of a mini-slot size, some padding is needed to make it so).

Packetization Interval, Encoding scheme	No Payload Header Suppression			With Payload Header Suppression		
	Payload size in Bytes	Overhead in Bytes	Data rate in Kbps	Payload Size in Bytes	Overhead in Bytes	Data rate in Kbps
10 ms, G.711	80	93	140.8	80	53	115.2
10 ms, G.728	20	93	102.4	20	53	64
20 ms, G.711	160	93	102.4	160	53	89.6
20 ms, G.728	40	93	57.6	40	53	38.4

Table 1: Voice Data Rate Variations

The choice of encoding scheme may depend on the voice application (e.g., conversational voice may use G.728 whereas fax and voice-band data may use G.711) or the degree of network congestion (G.711 may be used under light load and G.728 may be used under heavy load). Besides the varying data rates there are other reasons to consider a multi-rate voice model as well. One example is to be able to treat "intra-call" situations where both the called and calling parties use the same channel. In that situation, the "intra-calls" consume twice as much bandwidth but have half as much arrival rate compared to inter-calls whose two end-points use different channels. "Intra-calls" have been analyzed in [10,23].

Data Sessions

It is reasonable to assume that data session arrivals follow a Poisson process (see, e.g., [15,16]). However, the number of packets in a session has an arbitrary distribution that includes heavy-tailed distributions (e.g., Pareto or one of its modified forms). Also, there may be multiple file transfers within a session with some gap between successive file transfers. The individual packets in a session come at the peak rate that is allowed to each data user (except for the natural flow control in TCP during congestion). Modeling each session as TCP may slow down the simulation and so the impact of TCP is taken into account in an indirect and approximate way. During congestion, file transfer rates are reduced, packets are queued back at the cable modem and DOCSIS serves the various SIDs in roughly a round-robin manner. We assume that TCP instantaneously adapts to this reduced rate. The packet size during the file transfer is limited by the Maximum Transfer Unit (MTU) of Ethernet (about 1500 bytes). We designate these packets as *big*. In addition there are also *small* packets generated due to many reasons such as acknowledgment for downstream traffic, query, and control messages for initiating, maintaining or terminating sessions. The *small* packets arrive according to a Poisson process and they have an arbitrary length distribution bounded upwards by the MTU size and downwards by the smallest allowed packet size.

INTERACTION AMONG VOICE, DATA AND VOICE SIGNALING

Voice is typically given priority over other services over a certain portion of the available channel bandwidth using the unsolicited grant mechanism at an upstream channel and using priority queueing, weighted fair queueing or weighted round robin mechanism at a downstream channel. Therefore, in order to compute voice blocking we typically do not have to consider any other traffic types. In general a voice packet may have to wait for one large data packet transmission. This is typically not a significant issue over the downstream due to its large bandwidth and it may be addressed adequately over the upstream using the fragmentation feature of DOCSIS 1.1. The case of combined DOCSIS 1.0 and 1.1 operation will be addressed later. Also, there are interesting interactions among voice calls that need to be taken into account (e.g., several voice calls may use the same SID).

Delay of data packets and the throughput of data sessions are affected by the presence of voice since data essentially can only use bandwidth left over by voice. Proper engineering of voice traffic is needed to ensure acceptable data performance. Over the upstream, data performance also depends on the way the voice unsolicited grants are allocated since a data packet has to fit in the gap between the unsolicited grants. Since each data packet has to use the request mechanism to get a grant over the upstream, the data performance depends on the efficiency of handling collisions over the request region. The performance of data file transfer may be improved by using the piggybacking mechanism in which a field in the current packet is used as a grant request for the next packet. Finally, the data performance also depends on the maximum rate allowed to a data session and the number of data sessions present simultaneously. The rate available to a data session at a given instant is variable and TCP adjusts itself to the changing rate.

Voice signaling packets contend with data packets over the upstream until the unsolicited grants for the voice stream is established and special mechanisms are needed to ensure acceptable performance of signaling packets in the presence of large data file transfers. Over the downstream, standard QoS mechanisms may be used to ensure acceptable performance of voice signaling packets.

MODELING: SIMULATION VS ANALYSIS

We have developed a simulation model of DOCSIS 1.0 and 1.1 over the upstream that captures the unsolicited grant mechanism for voice transfer and request-based grant mechanism for data transfer. The model can also capture any type of arrival process and call/session duration distribution for voice calls and data sessions. For voice calls the model can also capture the delay/loss performance of adaptive de-jitter buffers. We have also developed several analytic models (faster to run compared to simulations) to capture various aspects of performance. Most of these models are based on Markov chains. Many of the models also have product-form structure resulting in fast computation even in the presence of many classes of voice and data traffic. In general the models assume Poisson arrival of sessions/calls and exponential session/call duration. However, in many cases the models are insensitive to call/session holding time distribution and the distribution of idle time of a voice/data source and so arbitrary distributions including heavy-tailed ones may be allowed. Besides the Markov-chain based models we have also developed linear and integer programming based models for optimally assigning nodes to CMTS (Cable Modem Termination System) cards. We provide below an example of Markov-chain based modeling. See [23] for other examples. Also, an integrated voice-data model with Poisson arrival of individual voice and data packets has been analyzed in [17-19].

Example Analytic Model: Voice-data Coexistence (Upstream or Downstream)

Let R_T represent the total channel bandwidth available to voice and data, i.e., after subtracting the bandwidth used for management and signaling in the downstream and management, signaling and request contention in the upstream. For the upstream we also assume that the fragmentation feature is used and the fragmentation overhead is accounted for in the data packets. Let $R \le R_T$ represent the total channel bandwidth available to voice. Let there be q classes of voice calls and s classes of data sessions. Voice call and data session arrival processes are Poisson and the number of packets in a call/session has an arbitrary distribution including a heavy-tailed one [15,16]. Voice calls are blocked if there is not adequate bandwidth, but once admitted, they are guaranteed the data rate r_i (for class i) (using the unsolicited grant mechanism in the upstream). Data sessions can use whatever bandwidth is not being used by voice but individual sessions also have a maximum allowed data rate r_{di}. Data sessions do not have any minimum guaranteed bandwidth and are not blocked. At a given time, if the sum of the data rates for all the voice calls plus the sum of the maximum data rates for all data sessions is less than or equal to R_T then all calls get their requested rates. If however, this requirement is not satisfied then the voice calls continue to get their rates but one or more data sessions get less than their maximum data rates. To consider the impact of this reduced data rate we consider a *bufferless* model and a *buffered* model. In the *bufferless* model it is assumed that the total duration of the data session is not affected due to this reduced data rate. There are two ways of interpreting this. One way is to assume that the excess data traffic is just lost as in Rate Envelope Multiplexing [20]. The other way is to assume that if the data rate is slower then the user transfers less data and keeps the duration of the data session the same. In the *buffered* model it is assumed that the total number of bytes transferred in a data session is unaffected by congestion and so the data session duration gets elongated.

Bufferless Model

We only consider the finite source version of data sessions and voice calls but the results can easily be extended to the infinite source case by taking appropriate limits. The model has product-form solution (see [3,4]) and insensitivity (except through mean) to both the distribution of session/call duration (which may be heavy-tailed) and the distribution of idle time duration of individual sources.

At first we consider voice calls only since they are unaffected by the presence of data sources. For class i of voice calls we assume that N_i is the number of sources, α_i is the average call arrival rate per idle source, β_i is the offered erlang per source in an infinite-server system, i.e., a system with no blocking, and $a_i = \alpha_i T_i$. Clearly, $\beta_i = a_i/(1+a_i)$ and $a_i = \beta_i/(1-\beta_i)$. The Steady-state probability vector $p(\mathbf{n})$ has a product-form solution given by:

$$p(\mathbf{n}) = g(R,\mathbf{N})^{-1}\prod_{i=1}^{q}\binom{N_i}{n_i}a_i^{n_i} \quad (1)$$

where the normalization constant $g(R,\mathbf{N})$ is given by,

$$g(R,\mathbf{N}) = \sum_{\mathbf{n}\in S(R,\mathbf{N})}\prod_{i=1}^{q}\binom{N_i}{n_i}a_i^{n_i},$$

$$S(R,\mathbf{N}) = \left\{0 \le n_i \le N_i,\, i=1,\cdots,q : \sum_{i=1}^{q} r_i n_i \le R\right\} (2)$$

The call blocking probability, P_{bi} for class-i voice calls is given by,

$$P_{bi} = 1 - \frac{g(R-r_i,\mathbf{N}-\mathbf{e}_i)}{g(R,\mathbf{N}-\mathbf{e}_i)} \quad (3)$$

where $\mathbf{e}_i$ is a vector with 1 in the *i-th* place and 0 everywhere else. The generating function of the normalization constant is given by:

$$G(z,\mathbf{N}) \equiv \sum_{R=0}^{\infty} g(R,\mathbf{N})\,z^R = \frac{\prod_{i=1}^{q}(1+a_i z^{r_i})^{N_i}}{1-z} \quad (4)$$

Recursive algorithms for computing g(R,**N**) may be developed from G(z,**N**) [12] and as in [5-7,13], highly efficient numerical inversion algorithms may also be developed. Similar algorithms with guaranteed minima and upper limits on the bandwidth usable by different call types have been developed in [11]. Recursive algorithms in the infinite source case have been developed in [8,9].

Next we consider data sessions and assume that class i of data sessions has mean duration T_{di}, N_{di} data sources, each idle source generates a new call at rate α_{di} and $a_{di} \equiv \alpha_{di}T_{di}$. Let $\mathbf{n}_d = (n_{d1},\cdots,n_{ds})$ represent the vector of number of data sessions of various classes and $p_d(\mathbf{n}_d)$ represent its steady-state probability distribution. Since there is no blocking of data sessions and the data session duration is unaffected by congestion, $p_d(\mathbf{n}_d)$ is given by the expression

$$p_d(\mathbf{n}_d) = \prod_{j=1}^{s}(1+a_{dj})^{-N_{dj}}\binom{N_{dj}}{n_{dj}}a_{dj}^{n_{dj}} \quad (5)$$

We define the probability of no data degradation, P_{ndd}, as the probability that all the data calls get their respective maximum rates and

$P_{dd} \equiv 1 - P_{ndd}$ as the probability of data degradation. In order to engineer a channel for combined voice-data service we use the voice call blocking probability as the performance measure for voice and the probability of data degradation as the performance measure for data. (Other performance measures such as the mean rate for data calls of class i may also be derived).

$$P_{ndd} = 1 - P_{dd} = \Pr ob.(\sum_{i=1}^{q} r_i n_i + \sum_{j=1}^{s} r_{di} n_{di} \le R_T)$$
$$= \sum_{\substack{\mathbf{n} \in S(R,\mathbf{N}) \\ (\mathbf{n},\mathbf{n}_d) \in S_T(R_T,\mathbf{N},\mathbf{N}_d)}} p(\mathbf{n}).p_d(\mathbf{n}_d) \quad (6)$$

Where, $S(R,\mathbf{N})$ is as in (2) and

$$S_T(R_T,\mathbf{N},\mathbf{N}_d) = \left\{ \begin{array}{l} 0 \le n_i \le N_i, i = 1, \cdots q; 0 \le n_{dj} \le N_{dj}, \\ j = 1, \cdots s; \sum_{i=1}^{q} r_i n_i + \sum_{j=1}^{s} r_{dj} n_{dj} \le R_T \end{array} \right\} \quad (7)$$

For small q, s, R, and R_T it is possible to compute P_{ndd} directly from (6) but otherwise we need to develop more efficient algorithms. Using (6), (5), (1) and (2) it can be seen that

$$P_{ndd} = \prod_{j=1}^{s} (1 + a_{dj})^{-N_{dj}} g_T(R, R_T, \mathbf{N}, \mathbf{N}_d) / g(R,\mathbf{N}) \quad (8)$$

where $g(R,\mathbf{N})$ is as in (2) and the second normalization constant is given by

$$g_T(R, R_T, \mathbf{N}, \mathbf{N}_d) =$$
$$\sum_{\substack{\mathbf{n} \in S(R,\mathbf{N}) \\ (\mathbf{n},\mathbf{n}_d) \in S_T(R_T,\mathbf{N},\mathbf{N}_d)}} \prod_{i=1}^{q} \binom{N_i}{n_i} a_i^{n_i} \prod_{j=1}^{s} \binom{N_{dj}}{n_{dj}} a_{dj}^{n_{dj}} \quad (9)$$

Next we obtain the generating function as

$$G_T(z_1, z_2, \mathbf{N}, \mathbf{N}_d) \equiv \sum_{R=0}^{\infty} \sum_{R_T=0}^{\infty} g_T(R, R_T, \mathbf{N}, \mathbf{N}_d) z_1^R z_2^{R_T}$$
$$= \frac{\prod_{i=1}^{q} (1 + a_i (z_1 z_2)^{r_i})^{N_i} \prod_{j=1}^{s} (1 + a_{dj} z_2^{r_{dj}})^{N_{dj}}}{(1 - z_1)(1 - z_2)} \quad (10)$$

We obtain $g_T(R, R_T, \mathbf{N}, \mathbf{N}_d)$ by numerically inverting Equation (10) using the methodology in [5-7], and then obtain P_{ndd} from (8) even for large q, s, R and R_T.

Buffered Model

This model will be explored in detail in a future paper and here we just consider the simple case of one class of voice and one class of data calls. Furthermore, we assume each class to have infinite number of sources, Poisson arrivals and exponential holding times. In general this model does not have a product-form structure and is sensitive to the entire holding time distribution (even though our simulation study, not reported here, shows that the sensitivity is often not strong). Let λ and λ_d represent the arrival rates for voice and data calls, T and T_d represent their mean holding times, and $\mu = 1/T$, $\mu_d = 1/T_d$. Also, let r and r_d represent the data rates for voice and data calls. Let n and n_d represent the number of voice and data calls at a given instant. The state space is given by

$$S(R) = (0 \le n \le \left\lfloor \frac{R}{r} \right\rfloor, 0 \le n_d < \infty) \quad (11)$$

The state vector (n, n_d) follows a continuous time Markov chain with transition rates as given below:

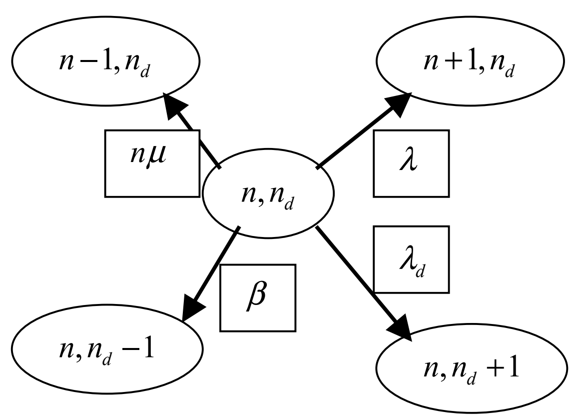

Figure 3: Markov Chain Transition Rates For the Buffered Model

Where,

$$\beta = \min\left(n_d \mu_d , \frac{R - nr}{r_d} . \mu_d \right) \quad (12)$$

Note that the second term within the minimum expression in (12) accounts for the rate reduction and associated elongation of data call holding time under congestion and is not there in the *bufferless* model. One difficulty in solving the Markov chain above is the infinite upper bound on n_d. In practice however an upper bound n_d^u can be placed on n_d above which state probabilities are too small or the data rate to data calls is so small that new data calls are refused. Once we solve the Markov chain, we can evaluate various performance measures such as the mean rate for data calls or the data degradation probability defined earlier.

NUMERICAL EXAMPLES

Some Generic Assumptions

- Upstream Channel:
  - 1.6/3.2 MHz RF, QPSK modulation, 2.56/5.12 Mbps.
  - Minislot size = 8 Bytes.
  - Contention and management regions each use 4% bandwidth on the average.
  - One request packet fits in a minislot
  - Map interval: Between 2 and 10 ms.
- Voice calls
  - 10 ms packetization.
  - G.711 with PHS.
  - Payload + overhead = 136 Bytes.
  - Poisson arrivals, exponential call holding time with mean 3 minutes.
  - 2.5 lines per subscriber, 0.14 Erlangs per line
- Data/Signaling packet overheads
  - All layer 3 and above overheads (IP and TCP or UDP) included in the packet size.
  - Layer 2 (Ethernet + MAC) overhead: 24 bytes.
  - If fragmented then fragmentation overhead: 16 Bytes.
  - Layer 1 (Physical) overhead: 34 Bytes.
- Data Sessions
  - Two types of packets
    - *Big*: 1500 bytes (Max allowed, File Transfers).
    - *Small :* 94 Bytes average, Coefficient of Variation 1 (Acknowledgments, Query, Control, Interactive, etc.)
    - 80% of the Packets are *Small.*
    - 80% of the Bytes Come from *Large* Packets.
  - *Big* packets come from file transfers with mean file size of 15 KBytes, Coefficient of Variation 2
  - Packet size: 64 Bytes (including Layer 3 and above overhead).
- Simulation Runs (in case of Simulation):
  - Each run captures 3 hours of operation with the first hour used for warm-up and statistics is collected only over the last two hours.
  - Each run is repeated 11 times. The first run is used to get a rough estimate of the 95th and 99th percentiles and the last 10 are used for actual computations including confidence intervals (not shown).

Voice-Data Integration: Hard Vs. Soft Partitioning, Impact of Packing of Voice Unsolicited Grants

We consider the following partitioning schemes among voice and data and packing schemes for voice unsolicited grants:

- Hard Partition: Voice and data/signaling use separate bandwidth regions with no sharing.
- Soft Partition: Data/signaling can use currently unused voice bandwidth. Two cases are considered
  - Random packing of voice unsolicited grants within each Map interval.
  - Pack unsolicited grants away from the data region and close holes within each Map interval. Closing of holes is done by potentially moving voice unsolicited grants

of existing calls as a call leaves (a scheme without closing holes has been analyzed in [14]).

We assume that fragmentation feature of DOCSIS 1.1 is used, upstream channel bandwidth is 1.6 MHz, at most 18 simultaneous voice calls (this uses up about 83% of bandwidth available to voice and data), 74 voice lines and data traffic being generated from 10 cable modems. Figure 4 below shows the performance of the three schemes. It is clear that the performance is substantially better for the soft partition schemes and among the two soft partition schemes, performance is better if all the voice unsolicited grants are packed at one side as one contiguous block thereby leaving one contiguous block for data and signaling.

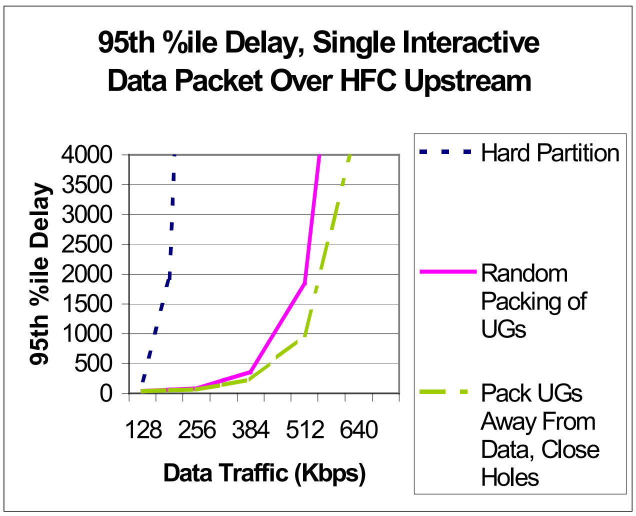

Figure 4

DOCSIS 1.0 Modems for Data and DOCSIS 1.1 Modems for Voice

DOCSIS 1.1 modems do not support fragmentation and so either the entire data packet or nothing can be transmitted in the gap left over by voice unsolicited grants. We at first assume that no jittering of voice packets is allowed to make some extra room for data packet transmission. Figures 5 and 6 show the performance of the two voice packing schemes under this condition and with MTU sizes of 500 and 1500 bytes respectively._ It is assumed that upstream channel bandwidth is 3.2 MHz and there are 135 data subscribers each being active for about 4.5% of the time. It is seen that the difference between the two packing schemes is much more significant compared to the pure DOCSIS 1.1 case in Figure 4.

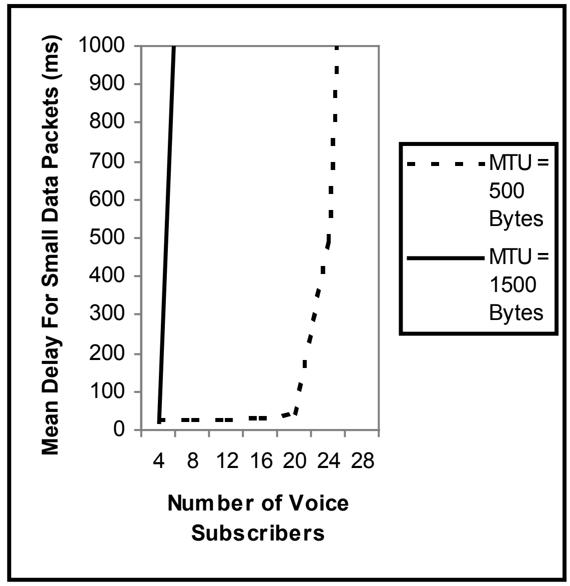

Figure 5: No Fragmentation Used, Pack Random

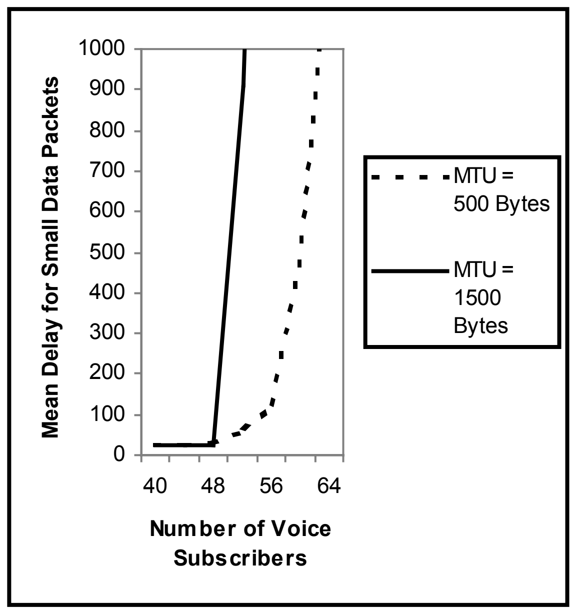

Figure 6: No Fragmentation Used, Pack UGs Away From Data and Close Holes

The performance of the "Pack UGs Away From Data and Close Holes" scheme can be further improved by allowing some delay jitters to voice calls. This is illustrated in Figure 7.

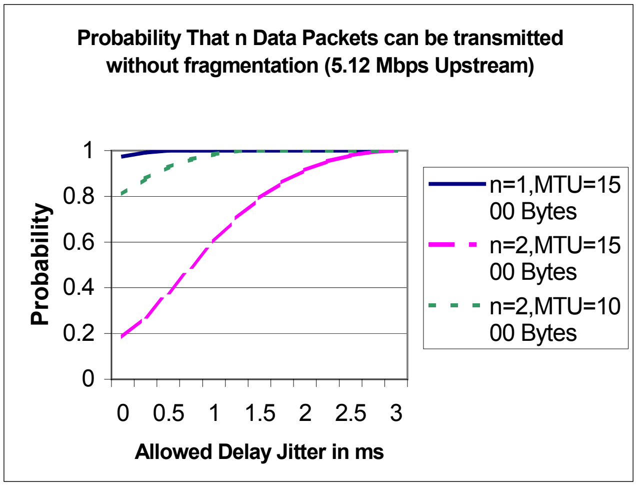

Figure 7

Voice Signaling Performance in the Presence of Data and Impact of Map Frequency

As mentioned earlier, the voice signaling packets ride with data over the HFC upstream until the unsolicited grants for the voice RTP stream is established. Figures 8-10 below consider three cases:

- Case 1: Data and signaling use the same SID.
- Case 2: Data and signaling use different SIDS but once requests of either kind arrive successfully at CMTS, they are treated at the same priority.
- Case 3: CMTS serves signaling at a higher priority (in addition to having separate SIDs for signaling and data).

Assumptions on voice and data are the same as in the first numerical example of voice-data integration and it is further assumed that signaling packets are 100 bytes long (including overhead) and only 2% of the (data + signaling) traffic is signaling (in terms of bytes). It is seen that signaling delay can be in the hundreds of ms to even seconds (depending on data traffic volume) in Case 1. This is quite unacceptable since the signaling delay may be encountered several times in the overall critical path of signaling. In case 2 the delay gets down to below 100 ms and in Case 3 the delay is only a few tens of ms even in the presence of heavy data traffic. The data traffic performance (not shown) is practically the same in all three cases due to the light volume of signaling. Therefore, it is preferable to use a separate SID for signaling and also give it higher priority compared to data at the CMTS.

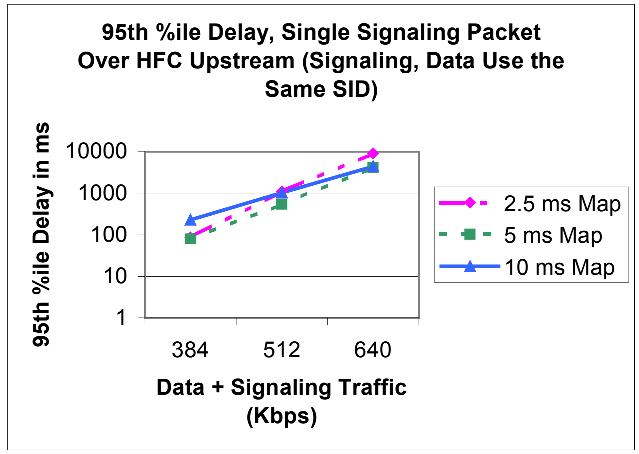

Figure 8

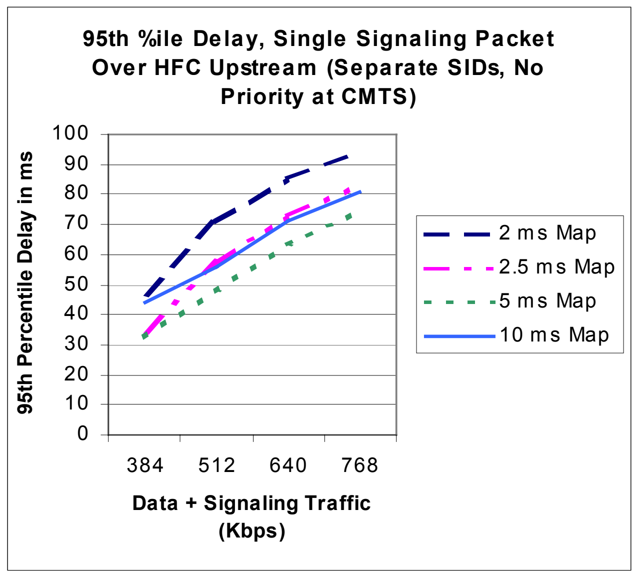

Figure 9

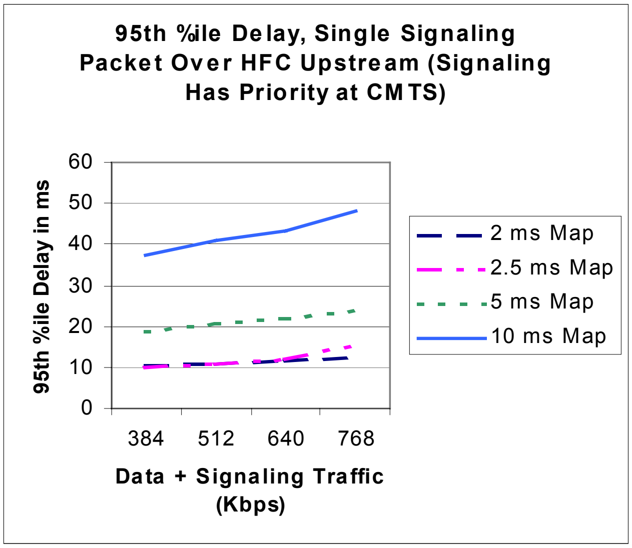

Figure 10

Voice-Path Latency Vs HFC Upstream Capacity

The voice path latency may be reduced by reducing the packetization interval but such a reduction has an adverse impact on the HFC upstream capacity. This tradeoff is shown in Tables 2 and 3. For voice-path latency computation it is assumed that with a 10 ms packetization, the typical/high estimates of round-trip delay at the cable access (or egress) is 40/55 ms for an end-to-end IP call and 70/90 ms if the call has to go over to the PSTN. Furthermore it is assumed that typical/high estimates of round-trip delays at PSTN access, PSTN backbone and IP backbone for a coast-to-coast call are 6/10 ms, 68/94 ms and 95/115 ms respectively. For HFC upstream capacity computation it is assumed that channel RF bandwidth is 1.6 MHz, 25% of bandwidth set aside for data/request/management/signaling, 0.5% call blocking probability, 30% take-rate for voice, efficiency of upstream channel pooling 10% less compared to full access and G.711 with PHS. Based on Tables 2 and 3 it appears that a reasonable compromise between low voice-path latency and high HFC upstream capacity would be to assume a packetization interval of 10 to 15 ms.

Type of Call	Packetization Interval			
	5 ms	10 ms	15 ms	20 ms
End-to-End IP	165/215	175/225	185/235	195/245
PSTN Backbone/Egress, Cable Access	134/184	144/194	154/204	164/214
PSTN Backbone, Cable Access/Egress	188/254	208/274	228/294	248/314

Table 2: Typical/High Round-trip voice-path latency for Coast-to-coast calls (in ms)

Packetization Interval	# of Simultaneous Calls Per Upstream	# of Households Passed (6 Upstreams, no Channel Pooling)	# of Households Passed (6 Upstreams and Channel Pooling)
5 ms	12	300	480
10 ms	17	500	716
15 ms	20	640	860
20 ms	22	720	956

Table 3: HFC Upstream Capacity

Impact of HFC Upstream Capacity on PHS, Compression, Modulation and Channel Width

The impacts are shown in Table 4. The assumptions are the same as in Table 3 except that no upstream channel pooling is considered and it is assumed that total available upstream RF bandwidth is 9.6 MHz, which can be partitioned into 1.6 MHz or 3.2 MHz RF channels. Also, both QPSK and 16QAM modulations are considered.

Encoding Scheme and Header Suppression	Capacity in # of Households Passed			
	Channel Width 1.6 MHz		Channel Width 3.2 MHz	
	QPSK	16QAM	QPSK	16QAM
G.711, No PHS	340	860	470	1030
G.711, PHS	500	1140	660	1360
G.728, No PHS	680	1420	830	1670
G.728, PHS	1240	2260	1440	2570

Table 4

A Methodology for Combining Nodes to a CMTS

Typically, a CMTS or each card of a CMTS supports a certain number of downstream and upstream channels. Using the methodology in Section 5 we can compute the number of voice and data subscribers that can be supported in each such channel and based on market penetration assumptions those numbers can be converted to the number of HHPs (households passed) that can be supported per CMTS card. Suppose in a given neighborhood there are *N* nodes and the *i- th* node has $H(i)$ households passed. The problem we address in this section is how to find an assignment of nodes to CMTS cards that satisfies the following:

- A CMTS card is assigned no more than SMAX (e.g., 4) nodes,
- A CMTS card is assigned no more than HMAX (e.g. 2500) HHPs

Optionally, for any feasible assignment of nodes to CMTS cards and for every card, calculate left-over HHP margins and the smallest one among them and then find an assignment that maximizes the smallest margin value. The rationale for doing this is to keep the maximal amount of idleness at each CMTS card for future growth. This is a "packing" type problem. We first need to estimate the number of CMTS cards, *C*, by rounding up $\sum_{i=1}^{N} H(i)$ /HMAX to the closest integer. The distribution of the *H(i)'*s and their Size compared to HMAX could result in breakage and require increasing the value of *C* until a feasible solution is found. In what follows we formulate a Linear Binary Program to solve this problem.

Variables:

- *X(i,c)* : assignment variable. Equals 1 when node 'i' is assigned to card 'c' and equals 0 otherwise.
- *y(c)* >= 0 : left over HHP margin in card 'c'
- w : A weight factor which assumes the value 1 for optimal solution and 0 for feasible solution

Constraints:

- $\sum_{c=1}^{C} x(i,c) = 1$ for all 'i' every node must be assigned to some card
- $\sum_{i=1}^{N} x(i,c) <=$ SMAX for all 'c' a card serves at most SMAX nodes
- $\sum_{i=1}^{N} x(i,c).H(i) + y(c)$ = HMAX for all 'c' a card serves at most HMAX HHP
- $y(c) \geq z$ for all 'c' all cards have at least a margin of z HHP
- z >= YMAX the minimum margin is at least YMAX

Objective:

- Maximize w*z

We implemented the model with AMPL [21] and CPLEX [22]. For small problems (e.g., C = 10) we can obtain both the feasible and optimal solutions very quickly by setting w = 1 and YMAX = 0. For large problems (e.g., C=70) we can still find the feasible solution quickly by setting w = 0 and an estimate for YMAX but the optimal solution takes too long to verify. With consistent unit changes, subscribers can be used in place of HHPs in this model. For illustration purpose, we present in Figure 11 an optimal solution to a 10-node 5-CMTS-cards problem.

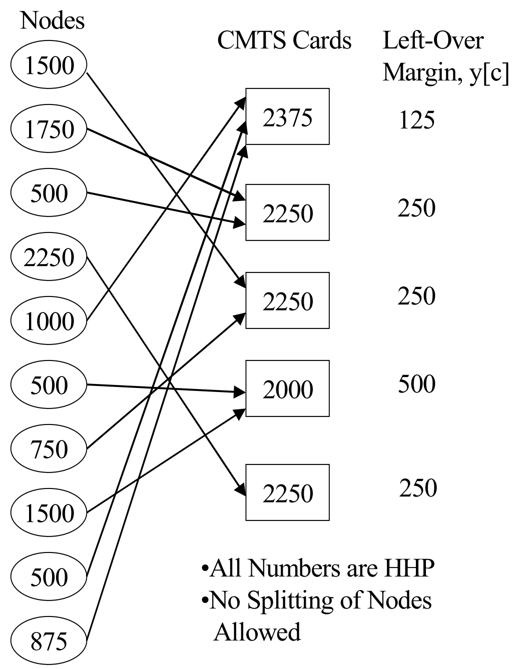

Figure 11. Optimal Assignment of 10 Nodes to 5 CMTS Cards

CONCLUSIONS

We have developed analytic and simulation models for the purpose of engineering, analyzing and controlling combined voice and data traffic over the HFC architecture. We show that the overall system performance and capacity are strongly affected by the modulation and encoding schemes, DOCSIS features such as fragmentation and payload header suppression, and the sharing technique between voice and data.

REFERENCES

[1] *Data-Over-Cable Service Interface Specifications, Radio Frequency Interface Specifications 1.0*, SP-RFI-I05-991105, November 1999, Cable Television Laboratories, Inc.
[2] *Data-Over-Cable Service Interface Specifications, Radio Frequency Interface Specifications 1.1*, SP-RFIv1.1-I05-00714, July 2000, Cable Television Laboratories, Inc.
[3] E. Brockmeyer, H.L. Halstrom and Arns Jensen, "*The life and works of A.K. Erlang*", The Copenhagen Telephone Company, Copenhagen 1948.
[4] F. P. Kelly, *Reversibility and Stochastic Networks*, Wiley, New York, 1979
[5] G. L. Choudhury, D. Lucantoni and W. Whitt, "Multi-Dimensional Transform Inversion with applications to the transient M/G/1 Queue," Annals of Applied Probability, vol. 4, pp. 719-740, 1994
[6] G. L. Choudhury, K. K. Leung and W. Whitt, "An Algorithm to Compute Blocking Probabilities in Multi-rate Multi-Class Multi-Resource Loss Models," *Advances in Applied Probability 27,* , pp. 1104-1143, December, 1995.
[7] G. L. Choudhury, K. K. Leung and W. Whitt, "An Inversion Algorithm to Compute Blocking Probabilities in Loss Networks with State-Dependent Rates," *IEEE Transactions on Networking*, Vol. 3, No. 5, pp. 585-601, October 1995.
[8] J. S. Kaufman, "Blocking in a Shared Resource Environment," *IEEE Transactions on Communications*, 29, pp. 1474-1481. 1981.
[9] J. W. Roberts, "A Service System With Heterogeneous User Requirements," In Performance of Data Commun. Systems and their Applications, ed. G. Pujolle, pp. 423-431, North Holland, Amsterdam, 1981.
[10] A. W. Jones, "Measurement of Non-Random Telephone Traffic with Special Reference to Line Concentrators," Proceedings of the 3rd International Teletraffic Congress, Paris, France, September 11-16, 1961.
[11] G. L. Choudhury, K.K. Leung and W. Whitt, "Efficiently Providing Multiple Grades of Service with Protection Against Overloads in Shared Resources," AT&T Technical Journal, Volume 74, No. 4, pp. 50-63, July/August, 1995.
[12] L.W.N. Delbrouck, "On the Steady-state distribution in a service facility carrying mixtures of traffic with different peakedness factors and capacity requirements," IEEE Trans. On Comm., 31, pp. 1209-1211, 1983.
[13] G. L. Choudhury and W. Whitt, "Q-Squared: a New Performance Analysis Tool Exploiting Numerical Transform Inversion," Demonstration at the 15th ITC, Washington, D.C., USA, June 23-27, 1997.
[14] W-S. Lai, "DOCSIS-Based Cable Networks: Impact of Data Packets on Upstream Capacity," 14th ITC Specialists Seminar on Access

Networks and Systems, Barcelona, Spain, April 25-27, 2001.
[15] V. Paxson and S. Floyd, "Wide Area Traffic: The failure of Poisson Modeling," IEEE/ACM Trans. On Networking 3, pp. 226-244, 1995.
[16] M. Crovella and A. Bestravos, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Cause," in: *ACM Sigmetrics* '96, 1996.
[17] M. Segal, "A preemptive priority model with two classes of customers, Proc. ACM/IEEE Second Symposium on Problems in the Optimization of Data Communications Systems," Palo Alto, Calif., October 20-22,1971, IEEE Cat. No. 71C59-C, pp. 168-174.
[18] S. Halfin and M.Segal, "A priority queueing model for a mixture of two types of customers," SIAM J. Appl. Math., Vol. 23, No. 3 November 1972, pp. 369-379.
[19] S. Halfin, "Steady-state distribution for the buffer content of an M/G/1 queue with varying service rate," SIAM J. Appl. Math., Vol. 23, No. 3 November 1972, pp. 356-363.
[20]
[21] J. W. Roberts (Ed.), "Performance Evaluation and Design of Multiservice Networks," Final report of COST 224 project, Office for oficial publications of the European Communities, Luxembourg, 1991.
[22] R. Fourier, D. M. Gay and B. Kernighan, "AMPL, A Modeling Language for Mathematical Programming," The Scientific Press, 1993.
[23] *Using the CPLEX Callable Library, Version 4.0*, 1995.
[24] G. L. Choudhury and M. Segal, "Cable Infrastructure to Carry Voice and Data – Performance Analysis, Traffic Engineering and Cable Head-end Node Design", 14th ITC Specialists Seminar on Access Networks and Systems, Barcelona, Spain, April 25-27, 2001.

2. Acknowledgements

We would like to thank Arun Guha, Wai Sum Lai, Bill Lester and Doug Nortz for their valuable input and comments.