DELIVERING PIXEL PERFECT

Dr. Robert L Howald, Dr. Sebnem Zorlu-Ozer, Dr. Nagesh Nandiraju
Motorola Home & Networks

Abstract

Operators are continuing to enhance their service mix with more personalized content, solutions that deliver the content to multiple screens, and doing so at accelerated deployment speeds. These objectives, among others, have driven plans to evolve the cable infrastructure towards an end-to-end IP architecture. Cable's IP pipe on the access network is, of course, the DOCSIS platform. However, the origins of DOCSIS were not developed with video services in mind. That has changed with DOCSIS 3.0. Nonetheless, supporting video requires the revisiting of traffic engineering principles used on today's DOCSIS access links.

Video over DOCSIS is expected to use H.264 encoding and variable bit rate (VBR) delivery, compared to legacy CBR MPEG-2 TS-based delivery and MPEG-2 encoding. In addition, novel adaptive streaming technologies offer intelligent alternatives to streaming models. Using a proven CMTS simulation tool, performance of video over standard DOCSIS links has been evaluated [2]. We extend these results for high and low action content, including the effects of peak capping, buffering, and CMTS configuration parameters on network performance. Quantifiable insight into the relationship between transmission losses and video performance will be examined. Finally, we will introduce adaptive bit rate technology into the model. These results will help operators understand the variables involved to traffic engineer their DOCSIS network for video services.

INTRODUCTION

The video service mix has gradually grown over the years in terms of technology, complexity, and consumer offerings – VOD, PPV, SDV, MPEG-4, OCAP, HDTV. The momentum of this march to video services paradise was jolted when a key crossroads occurred, as shown in Figure 1.

Suddenly, "HSD" went from meaning "High-Speed Data" to standing for "Heckuva Streaming Demand." Of course, this stage of data speed evolution represents an essential "must have" for the cable IP pipe to be considered as a means for delivery of video content. Figure 1 puts the inevitable into pictures, identifying that crossroads in time when high quality video rates became low enough that the increasingly fat data pipe could effectively deliver it to residential subscribers. An important point to make on the topic of cable IP video is that, in the context of this paper, we are referring usually to MSO-owned video assets, as opposed to over-the-top providers.

Why the fuss over IP delivery given the cost-effective infrastructure in place? There is no single answer, but instead a list that, when taken as a whole, makes a compelling case for migrating from purpose-built video system architectures to an all-IP architecture. Operators have routinely described these perspectives in many conference sessions and industry events, where key technologists espouse their views on when, why, and how.

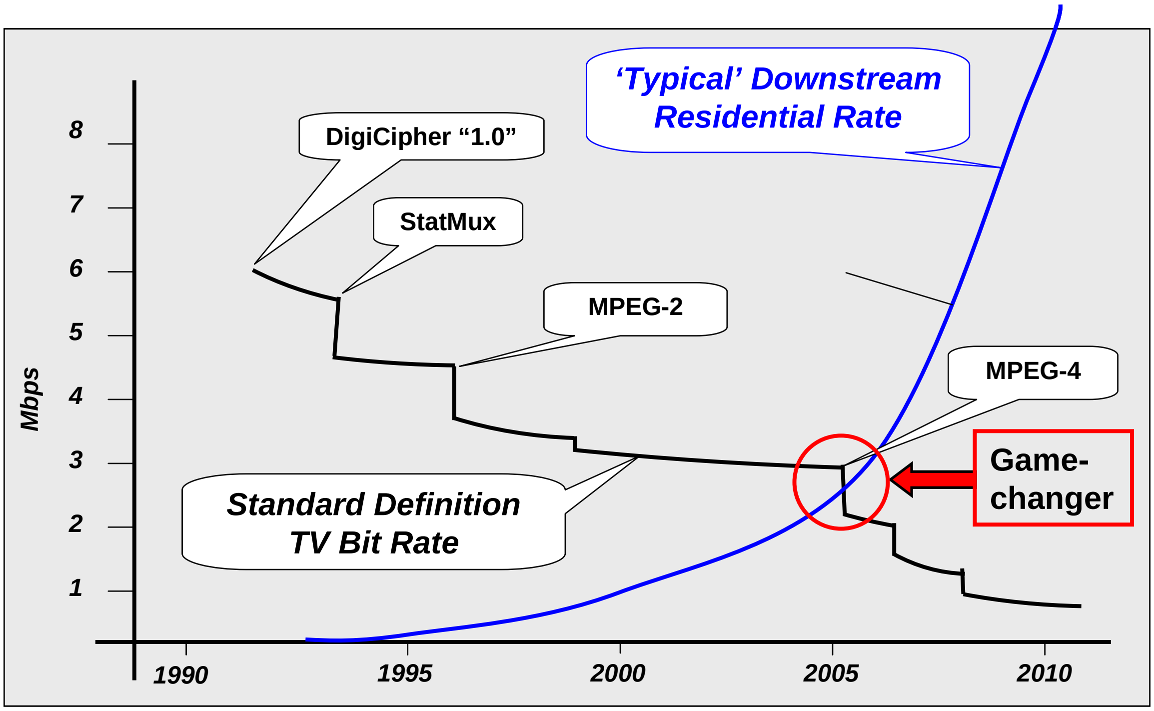


Figure 1 – Downstream Internet Speeds vs Digital Video Requirement

Typically, the reasons involved include readily enabling the multi-screen experience, compatibility with mature IP home networking technologies and initiatives, lower cost CPE, software-based security, enabling future alternative access networks, and closing the last loophole in E2E IP delivery, which is video delivery over HFC. This is expected to lead to improving the velocity of new services delivery and associated OPEX savings in the long run.

There are other obstacles besides the substantial legacy investment to achieving a full migration, one of which is new bandwidth. However, in general, there is quite a bit of underutilized downstream capacity. And, there are many techniques, traditional and not, to go about extracting it that can be deployed as traffic demands continue to increase [1].

While "how to" discussions take place and bandwidth expansion activities continue, a final important "how to" remains: how to system engineer the access edge for IP video. The significance with which video service affects traffic parameters and ultimately bandwidth occurs primarily in two ways:

1) The pure volume of bits-per-second required for video streams
2) The concurrency of use factor for video services vs browsing-based services for HSD

It is simple to show that video concurrency rates of 5-10% (VOD-like parameters) has significant impact on HSD bandwidth requirements, when considering that 1% or less is a typical data oversubscription rate.

PREVIOUS MODELING - SUMMARY [2]

Model Description

OPNET™ CMTS Model

In [2], a key modeling tool was developed for analyzing video over DOCSIS performance. It is the basis for the results presented there, and is leveraged and extended in this paper to further develop and refine video over DOCSIS performance. The model is based on a DOCSIS model using OPNET™, version 14.5. A simple reference scenario is shown in Figure 2, where a CMTS serves a set of homes with cable modems connected to subscriber equipment.

Modeling Input Stimuli

A large bulk of the modeling research is based on volumes of traces captured and made publicly available, and which can be easily be imported to the model as stimulus. Traces from a video clip library at http://trace.eas.asu.edu and http://trace.kom.aau.dk were used [4]. A brief description of what is encompassed in these online libraries is discussed in [2]. Generally, there are volumes of CIF (352x288) and HD traces across a range of PSNR and quantization settings. As pointed out, lower resolution formats such as CIF and VGA – common for smaller screens – tend towards a higher peak-to-average and thus represent conservative examples from a modeling perspective. We choose from these clips only the high video quality samples (PSNR of 40 dB or greater). The associated quantization parameters have the effects of creating higher rate CIF streams, representing values close to cable SD rates for H.264 (MPEG-4 AVC) encoding.

In addition to the streams above, some clips captured by Motorola were mixed in, as will be seen in the tables that follow which list the streams. Finally, in some cases, H.264 Scalable Video Coding (SVC) clips were used where it helped fill a wideband channel to exercise it at high utilization. Like CIF, SVC also has the property that it tends to aggravate peak-to-average variation, or coefficient of variation (CoV).

Summary of Key Results

This paper builds on the results of [2], so we will briefly summarize some of the key findings from those simulation examples.

A simple “static” gain model was created to point out the potentially large variation of bandwidth efficiency over CBR delivery based on content mix. Table 1 shows the range of efficiency “gain” of VBR – or, more accurately, adjustable CBR – under a very simple, illustrative, assumption of two video classes and MPEG-2 encoding. Assuming a 3.75 Mbps CBR system of 40 programs (four bonded channels of DOCSIS 3.0), and gain made available by allocating 2.50 Mbps to the “easy” programs, there are bits freed up to add more channels. Thus, “easy” programs offer 33% savings to spend elsewhere. For a mix of easy and hard channels that exist, and a desire to add new channels, also of each type, Table 1 shows the effective gains of this scheme, pointing out the dependency on the content type.

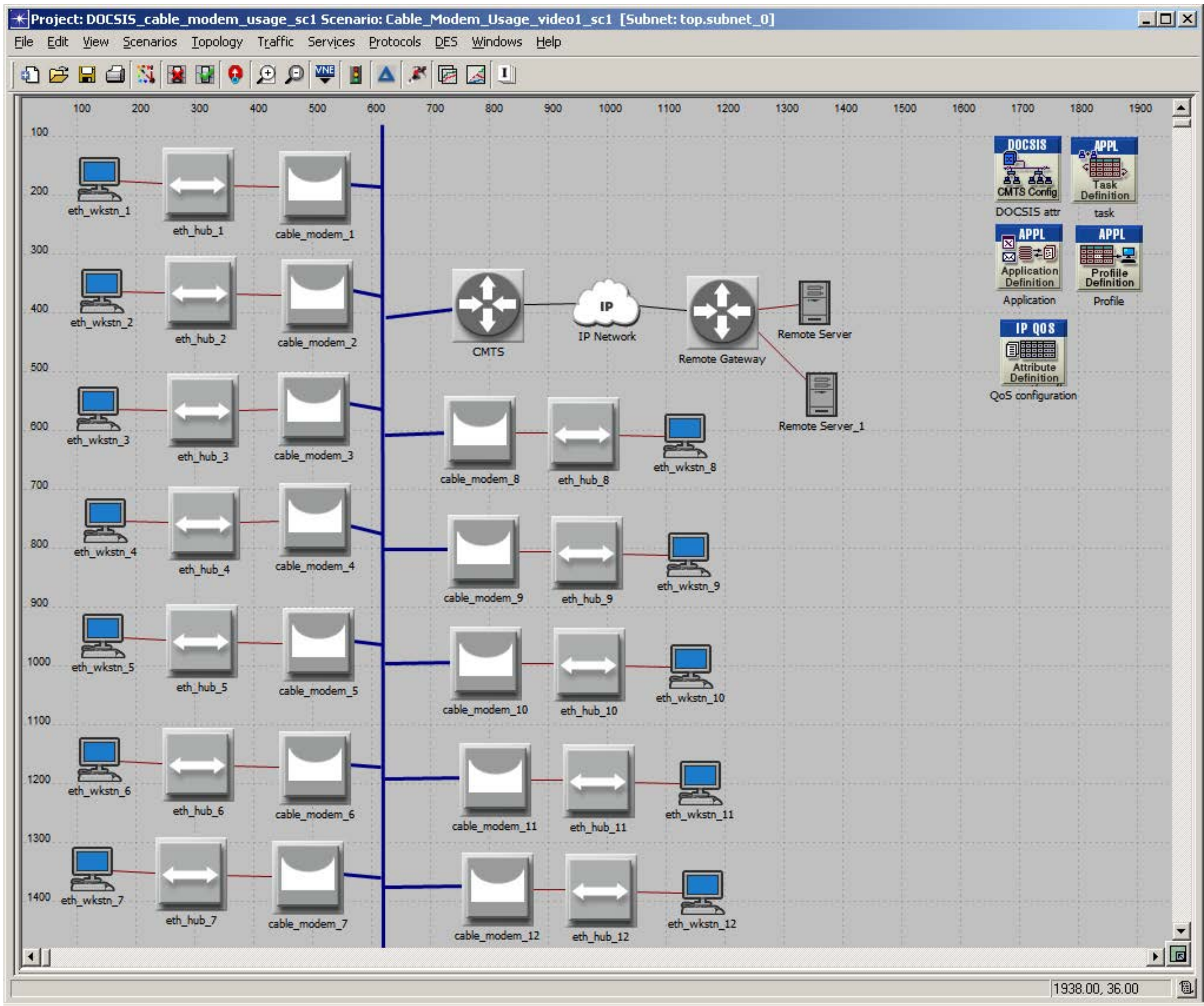


Figure 2 - Sample of a Simulation Scenario Using *OPNET™*

Table 1 – Efficiency Gain – Two Classes Example (Easy / Hard)

Added Programming Mix	Existing Programming Mix				
	70/30	***60/40***	***50/50***	***40/60***	***30/70***
70/30	30.4%	26.1%	21.7%	17.4%	13.0%
60/40	29.2%	25.0%	20.8%	16.7%	12.5%
50/50	28.0%	24.0%	20.0%	16.0%	12.0%
40/60	26.9%	23.1%	19.2%	15.4%	11.5%
30/70	25.9%	22.2%	18.5%	14.8%	11.1%

"Easy" = 2.50 Mbps
"Hard" = 3.75 Mbps

We see that the range of gain varies by nearly three times (11.1% to 30.4%) based on this reasonable range of content mix.

Channel Utilization and VBR Efficiency

Table 2 summarizes the comparison of existing research characterizing H.264 with the simulations described above in terms of percent channel utilization. This analysis, drawn from the same content pool, was described in detail in [2]. There is close agreement between analysis and simulation results, for both single channel and bonded, wideband channel models.

Table 2 – Simulated Utilization vs. Calculated [2]

	Supported Load	Overloaded
Analysis Min	55%	
Analysis Max	71%	
Analysis Avg	62%	
Simulated 1 QAM	63%	72%
Simulated 4 QAM	74%	76%

The simulation results were translated to VBR gains based on DOCSIS scheduling under a particular, typical configuration, a reasonable user buffering limitation, a packet loss threshold, and a factor for the grooming and multiplexing imposed on the streams prior to reaching the edge device due to standard video processing operations. Resulting estimates of VBR gain showed a range of 9-37% increased efficiency, depending on content mix and channel size (single vs four bonded channels).

The above summary of the analysis in serves as a useful baseline to further examples.

NEW SIMULATION RESULTS

DOCSIS 3 Channel Bonded SD + HD Mix

Additional simulations were performed on the mixed-resolution scenarios to quantify further the conclusions about the effects on bandwidth efficiency, and to further exercise system variables under typical configurations. The results of Table 2 indicated that for four bonded channels, 74% capacity utilization was achieved, while 76% utilization caused packet drops at a rate greater than the 1e-6 threshold chosen. In that model, the CPE buffer was fixed at 100 msec, putting a larger burden on the CMTS scheduler to process and deliver the video payloads efficiently without any statistical information to support network admission or congestion management.

This same HD + CIF content line-up was used as a starting point and modeled while making adjustments to network variables. If, for example, we allow the CPE buffer to increase to up to 500 msec – about the maximum that can be considered before other issues come into play – the model shows that additional streams (or higher rate streams) can be added, increasing the utilization efficiency. Measuring utilization efficiency as the overall mean rate of streams to throughput capacity, the channel utilization can be taken up to 78%, or about 6% additional gain over what was derived in [2]. This 78% efficiency can be held as well using a 400 msec CPE buffer, but in this case only if a maximum rate cap is enforced at 10.5 Mbps per stream. This would impact about half of the HD streams, and a few by a percent reduction that would be anticipated to be noticeable, particularly if sustained. With respect to VBR efficiency, the four channel bonded case relative to [2] was improved, resulting in efficiency gains varying from 24%-40%.

Figure 3 shows this truncated received traffic and the injected traffic when the CPE buffer was limited to 200 msec, an increase in buffer size, but still inadequate to accommodate the added traffic without a further increase.

Figure 4 and Figure 5 show the queuing and capping behavior that lead to mitigation of the congestion issue. The queuing delay can be outlasted as shown, and described above, with the appropriate buffer size. The impact of capping at different rates (10.5 Mbps, 11.0 Mbps, and 12.0 Mbps) is shown in Figure 5, where 10.5 Mbps draws the traffic level beneath the aggregate needed to avoid lost packets.

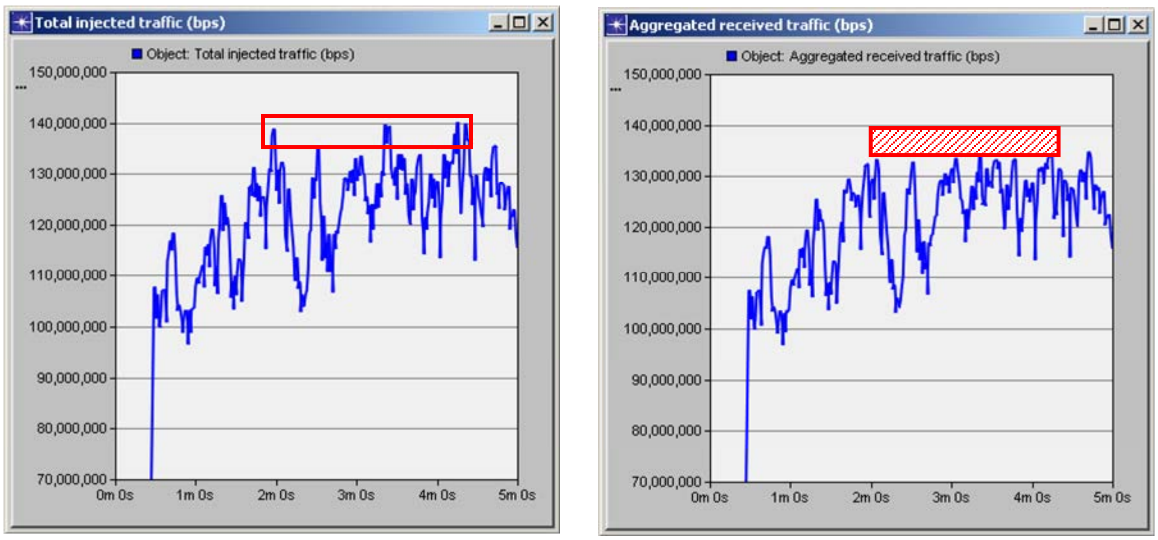


Figure 3 – Injected vs Received: Overloaded & Clipped Video Traffic

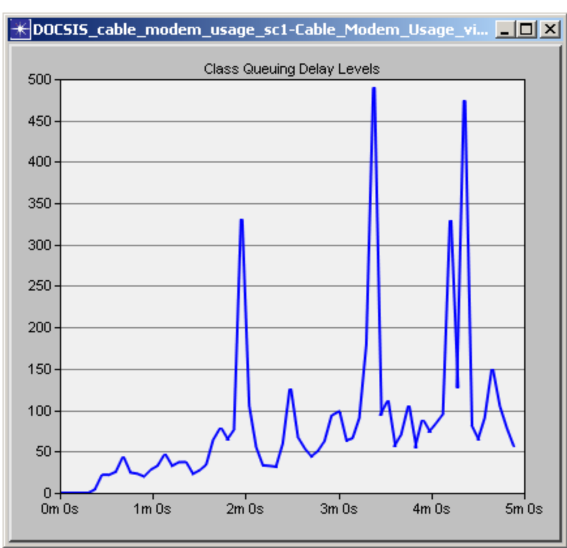


Figure 4 – Delay – Buffer Size Impacts

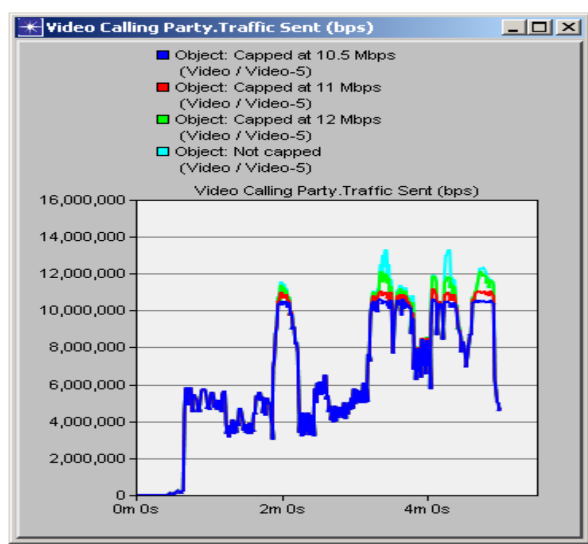


Figure 5 – Peaking-Capping Effects

Channel Efficiency: Low vs. High Action

To gain insight into the video content type dependency to efficiency, cases were run comparing sets of low-action and high-action content, in this case comparing single QAM carriage for HD – the least effective use case from a statistical multiplex perspective. While not a column in the Table 1, note that the efficiency gain of 100% "hard" content would be zero in the context of how Table 1 was derived. There would be no streams upon which an adjustment downward would be considered acceptable. Such is the case with the all high-action HD content simulation (Mean = 5.8 Mbps, pk-avg, sum basis = 2.34). Under this stimulus, drop-free transmissions occur when supporting four HD programs, which is essentially the expected CBR equivalent of HD/QAM for high action content using MPEG-4, and assuming an average 50% encoding gain. The utilization efficiency for this 4-HD program case was about 60%. This is in line with single-QAM efficiencies from prior simulations and shown in Table 2, despite the smaller statistical basis due to low stream count. This is likely due to the relatively well-behaved peak-to-average of the content mix (2.34) noted above.

Mixing in low and high action HD content, at what can be considered simplistically as 50/50 "hard" vs "easy," six streams of HD were fit within the single QAM. This represents a 50% "gain" in video programs compared to the "hard only" case, but roughly the same utilization efficiency. In this case, the slightly larger statistical basis leads to no better utilization efficiency than the high action case above. This is, again, likely because of the peak-avg behavior, which for the six-stream HD multiplex is higher than in the all high-action case.

The additional stream gain does compare favorably to Table 1, although not at first glance. Table 1 is based on a specifically chosen standard definition (SD) ratio that states that the CBR rate for "hard" content is established at 50% higher than the rate for "easy" content. For HD, at four programs/QAM, we would consider a CBR (considering overhead) of about 9.5 Mbps at MPEG-4 as a reasonable over-provisioned rate. Now, consider the six streams in the example shown in Table 3, and note the "easy" content – traces 14, 11, 1. The average of this set is about 2.2 Mbps, and the peak-to-average is nearly the same as the prior 4-stream example. Using the same relationship of average, peak, and allocation, this would establish "easy" HD at about a 3.70 Mbps (i.e. also about 60% utilization).

Table 3 – Mixed HD Content on a Single QAM

	Mean Rate (Mbps)	Peak Rate (Mbps)
13hd (Motorola trace) / (trace 1)	3.53	4.68
sony720_G12B2FxT22 / (trace 5)	6.50	13.87
Mars –segment 1 / (trace 6)	3.46	12.72
Mars – segment 3 / (trace 8)	6.25	16.20
Horizon – segment 1 / (trace 11)	1.64	5.63
Horizon – segment 4 / (trace 14)	1.50	6.52

To compare to Table 1, we need to begin, for example, with a 50/50 "Existing Program Mix" of "easy" and "hard." As such, assume the initially configured 50/50 CBR HD being composed of traces 1, 5, 8, 11. Remaining from the six are now one "hard" and one "easy," which means 50/50 also for the columns labeled "Added Program Mix." The analogous Table 1 column says that a 50/50 existing mix and a 50/50 added mix translates to a 20% stream count gain. However, seeing that our CBR hard-to-easy ratio is closer to 2.6:1, instead of 1.5:1, Table 1 instead becomes Table 4 below.

Going from 50/50 existing, to adding 50/50 with the savings from CBR, we in fact would expect 44% gain, for a total of 5.8 streams – nearly 6 streams. Of course 6 streams means adding two more, as this simulation has shown to be accurate. Looked at another way, the precisely 50/50 new stream case occurs when the existing ratio is somewhere between a 50/50 and 60/40 hard-to-easy ratio.

Finally, consider the case of only *low* action HD. The model uses the relevant subset of the HD traces used in [2], plus some not previously used traces to build up enough low-complexity content to fill a channel. Table 5 shows the line-up used for this example.

Table 4 – Efficiency Gain, HD – Two Video Classes Example

Added Programming Mix	Existing Programming Mix				
	70/30	***60/40***	***50/50***	***40/60***	***30/70***
70/30	74.6%	64.0%	53.3%	42.6%	32.0%
60/40	67.4%	57.8%	48.2%	38.5%	28.9%
50/50	61.5%	52.7%	43.9%	35.2%	26.4%
40/60	56.5%	48.5%	40.4%	32.3%	24.2%
30/70	52.3%	44.8%	37.4%	29.9%	22.4%

"Easy" = 3.7 Mbps
"Hard" = 9.5 Mbps

Table 5 – Low Action HD Line-up

	Mean Rate (Mbps)	Peak Rate (Mbps)
13hd (Motorola trace) / trace 1	3.53	4.68
06hd (Motorola trace) / trace 3	4.20	4.44
Mars –segment 1 / trace 6	3.46	12.72
Horizon – segment 1 / trace 11	1.64	5.63
Horizon – segment 2 / trace 12	1.55	2.93
Horizon – segment 3 / trace 13	1.57	2.77
Horizon – segment 4 / trace 14	1.50	6.52
Blueplanet – segment 1 / trace 51	1.7	12.2
Blueplanet – segment 2 / trace 52	1.9	8.01
Blueplanet – segment 3 / trace 53	2.03	9.64
Blueplanet – segment 4 / trace 54	2.18	7.18

In this case, 10 HD programs were able to be multiplexed in a single QAM channel, and an 11th channel is nearly able to be added. Drop-free transmission was possible for 11 channels, but only if the CPE buffer was allowed to exceed the maximum allowed by our definition (500 msec) at 750 msec. However, with bit rate capping at 7 Mbps (4 traces impacted, 2 with reservations), the buffer was able to be held within 500 msec. Mitigation of the peak excursions is shown in Figure 6.

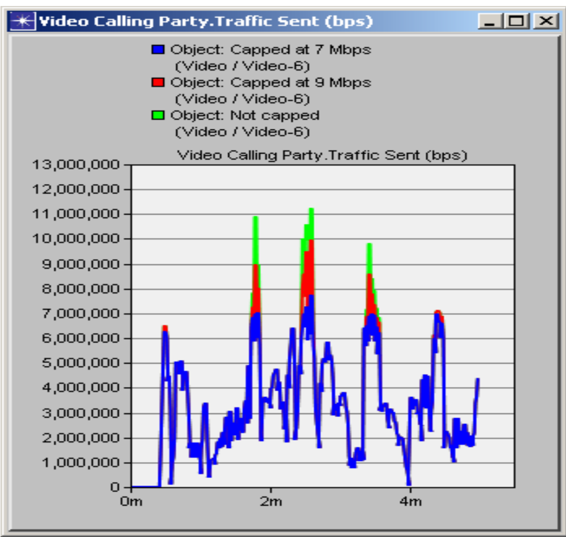


Figure 6 – Impact of Capping on Peak Excursions

A simple comparison to Table 1 and Table 4 can be made without creating a new table of possibilities. For 100% hard content, as discussed, we have four programs. If "easy" programs are 2.6 times as efficient, then we should have 2.6 times as many programs, or 4 x 2.6 = 10.4 programs, when all HD is low action. Indeed, we have shown that 10 streams are obtained, and almost 11.

On a broader basis, the ability to stream anywhere from 4 to 10 HD channels on a single QAM, depending on content type, again points out the high dependency of bandwidth efficiency to content type. The observed gains vary from -33% to +67% stream count efficiency if we consider as the baseline the mix of 6 HD streams, and consider that 4 streams fit when content is all high action, and 10 streams fit when the content is all low action.

<u>CMTS Configuration for Video Traffic</u>

It has been discussed often how video traffic characteristics differ in important ways than web browsing traffic. In general, video traffic is characterized by longer packet sizes and a more consistent rate of arrival. As such, the way a CMTS is configured for a voice + data mix is sub-optimal for how it might be configured in video-only mode. However, video frame size statistics are very complex, and video is much less tolerant of any issues in delivery. A mixture of video, voice, and data would be more complex still.

For modeling purposes, we will again consider the simple case at this point – assume that a four channel-bonded downstream is supporting video traffic only, a likely scenario initially for an MSO rolling out a managed IPTV service. The two primary mechanisms of packet drop are overflowing the transmit buffer, and excess delay that does not support buffer margin allocated on the receive side. Delay in the transmit buffer that is nearing the limit of time-to-live (TTL), and is determined unlikely to make it to the CPE in time, can also be dropped so other packets can be serviced, creating a secondary transmit-side packet loss scenario.

We choose the HD-only program line-up shown in Table 6, consisting again of segments of the trace library in [4] and Motorola-created segments. Some of the streams encodings are SVC, in order that the downstream channel would be filled at or close to its expected utilization for comparison and statistical purposes.

Table 6 – HD Streams on DOCSIS 3.0 Downstream

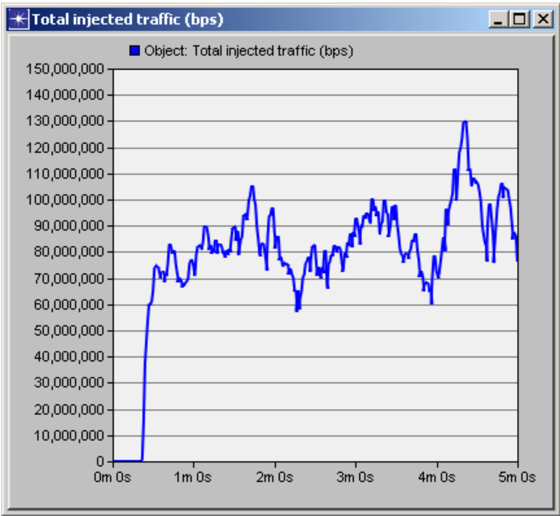
	Mean Rate (Mbps)	Peak Rate (Mbps)
13hd (Motorola trace)	3.53	4.68
05hd (Motorola trace)	9.64	15.25
06hd (Motorola trace)	4.20	4.44
sony720_G12B2FxT22	6.50	13.87
Mars –segment 1	3.46	12.72
Mars – segment 2	5.02	20.94
Mars – segment 3	6.25	16.20
Mars – segment 4	5.11	12.45
T2720_G12B2FxT22	5.43	12.04
Horizon – segment 1	1.64	5.63
Horizon – segment 2	1.55	2.93
Horizon – segment 3	1.57	2.77
Horizon – segment 4	1.50	6.52
Blueplanet1080_G16B3c – segment 1	2.98	6.83
Blueplanet1080_G16B3c – segment 4	4.68	17.7
Blueplanet1080_G16B3c– segment 3	5.26	13.3
Blueplanet1080_G16B3c – segment 2	5.28	11.98
Transporter2_1080_G16B3c – segment 1	10.24	29.84

This mix fits comfortably in the DOCSIS 3.0 channel with no packet loss issues, using a CPE buffer size of 300 msec. A buffer size of 200 msec works if rates are capped at 22 Mbps, which impacts only the extremely dynamic *Transporter 2* clip. While the vast majority of the time there is acceptable delay to the end user (we do not account for phy layer delay in our simulations), a portion of the aggregate traffic experiences a spike that dominates network performance during our 5-minute segment. The average transport packet delay experienced at this peak of the aggregate transmission burst is about 160 msec, and thus the reason the buffer size moving from 200 msec to 300 msec can make a difference. The rest of the sequence generally stays below 40 msec.

Rate Limiter Adjustments & Peak Bursts

In order not to vary CPE buffer size, which may exist in the field or be otherwise fixed due to memory limitations (such as to 200 msec), we can alternatively modify configuration parameters of the CMTS to accommodate the expected increase in burst size of video frames and avoid packet loss. The model also allows us to see what happens as streams align themselves unfavorably – peak bursts aligned – such that even this modest traffic load from a utilization standpoint (about 54%) can encounter congestions. We can then compare with tools available to mitigate this scenario. Let's examine these two scenarios.

Figure 7 shows two cases of traffic injection. On the left is aggregate injected traffic aligned through just the random time selection of segments from the library, versus on the right where they are slid around to create the most stressful network condition at roughly the 4.5 minute mark. The right-hand side of Figure 7, where the peaking is deliberately aligned, also shows the received aggregate traffic, pointing out the region of a lost burst of packets.



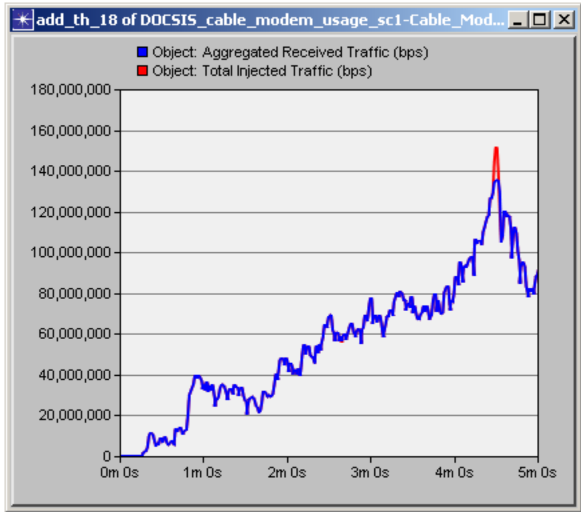


Figure 7 – Injected Traffic – Random (L) and Misaligned (R)

Now consider Figure 8, which shows the worst case packet delay observed between the two cases of stream alignment. In the top figure, we have increased an internal rate limiting function to provide a higher peak, so as to not allow a large burst to hit a stop sign on the way to the scheduler, meaning less opportunity for a large frame to be truncated. The result is that the *maximum* delay is dropped to about 65 msec (from the 160 msec avg in the section introduction).

In the lower figure of Figure 8, it is immediately obvious why this scenario could cause packet loss. We see delay exceeding at least 400 msec at the peak burst, even though the bulk of the time the network delay performance is quite sufficient. As would be expected given the perfect misalignment, a spike of traffic at the 4.5 minute mark is the cause of congestion and loss. While this example was deliberate misalignment, it was done to replicate a potentially realistic scenario for unmanaged streams, given that these are only five minute segments. Such a scenario becomes statistically more likely when the five minute span is scaled over by long periods of time and content mixes.

Quantum & A-Priori Knowledge

Another scheduler parameter that can be used to take advantage of the more predictable range of input traffic from video is the round robin quantum. In addition, some knowledge about the stream, either as a stored asset or gathered in near-real time, can be used to manage congestion and performance. We examine these cases here.

Figure 9 shows a comparison of increasing the quantum from several maximum Ethernet frames to the order of 100 Kbytes. The latter reduces the maximum observed delay, the important parameter in order that we do not drain buffers and ultimately starve decoders, by about 33%. This quantum size is enabled by the single class of service, which is also supporting only a single service type. Thus, only rounds of service are lost at the benefit of a high probability of fully servicing. But, as a single class and service type, there is decreased concern for the unfairness this can cause to smaller bursts. The former (the top half of Figure 9) results in a maximum observed delay of almost 100 msec, compared to about 65 msec when anticipating video-only traffic.

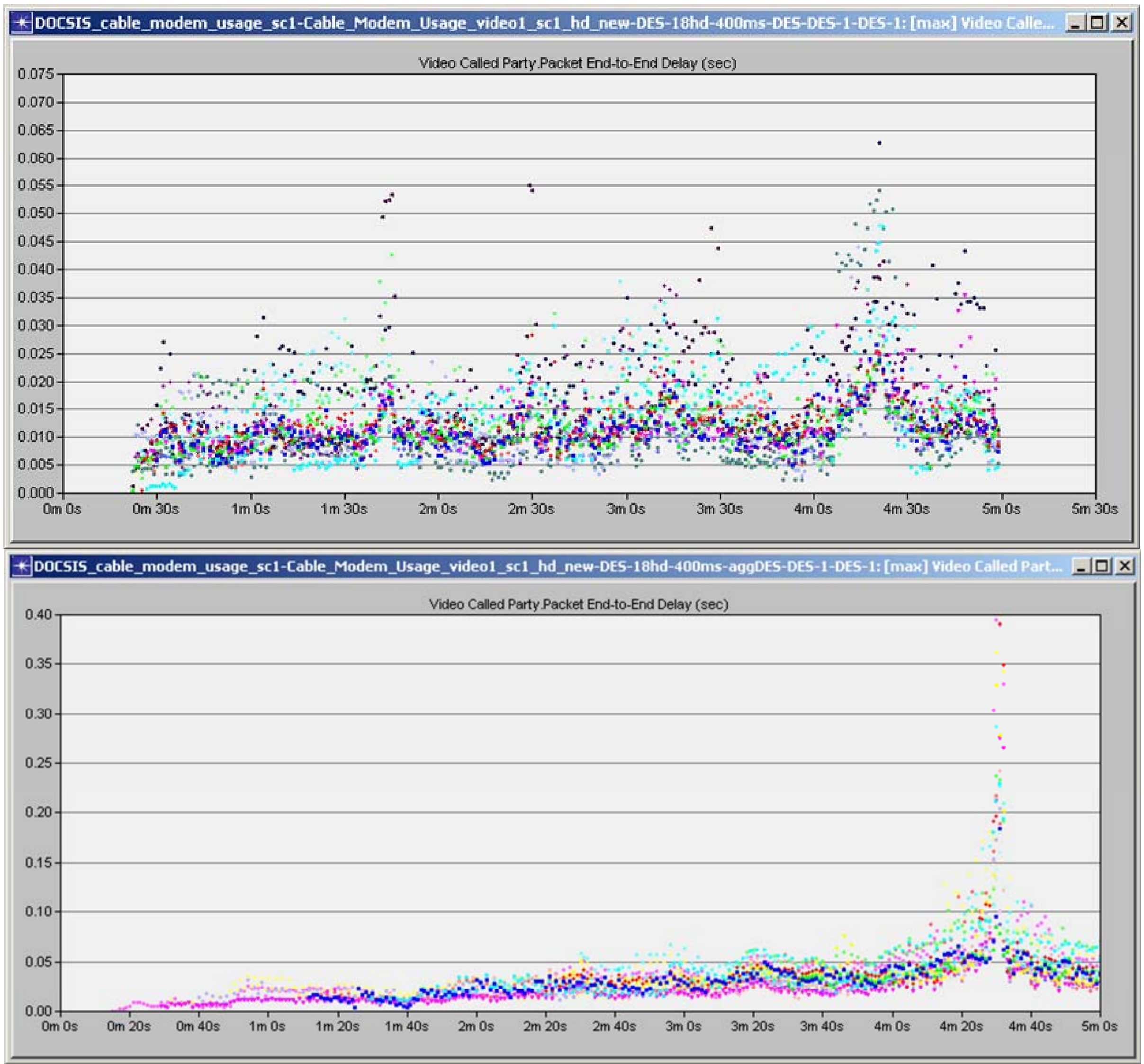


Figure 8 – Packet Delay – Decreased Rate Limiting (T) and Misaligned (B)

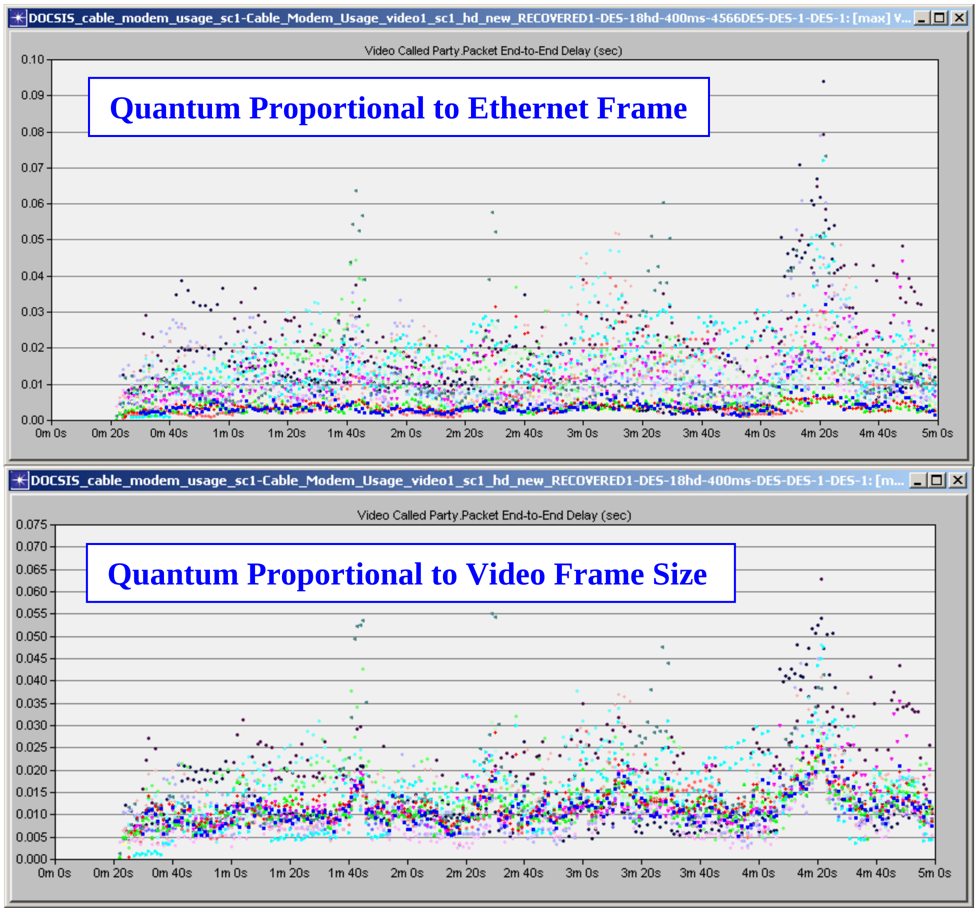


Figure 9 – Packet Delay vs Quantum: Scaled to E-net Frame vs Video Frame

Ideally, video streams would be accompanied by an array of metadata advertising their statistics. And, for stored assets, there is nothing in principle from comprehensively characterizing a stream statistically. However, the statistical variation for video from stream to stream and within a stream is large, and aggregation allows the law of large numbers to come into play. Thus, the added complexity beyond first and second order moments is typically not undertaken.

Some simple constraints, such as maximum size frame and peak arrival rates, can go a long way towards a deterministic network response (or more accurately a deterministically bounded response), if the constraints themselves can be guaranteed – which is a big "if." An example is shown in Figure 10, where we have added to network stress by once again choosing the worst case alignment of streams shown in Figure 7 (R) and Figure 8 (B). Though phase aligned for peaks, we have in this case capped the maximum frame size, set the quantum according to it, and assumed that we know the servicing rate (internal) and arrival rates (external). With that set of constraints, the delay becomes an arithmetic problem. That is, if we know how large the packets can be, how frequently they arrive, and how quickly we can service them, it is straightforward to calculate what's in a queue and the delay in servicing that queue, which is key to delivering on time and under budget. Figure 10 shows the precision for which we can assure a particular behavior under a set of assured constraints for a simulated and calculated queue size.

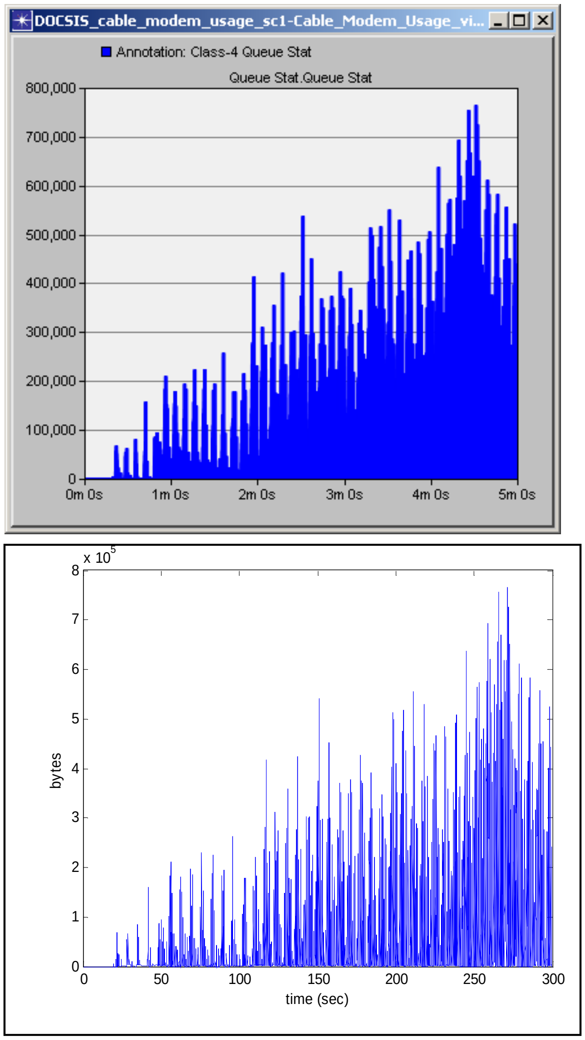


Figure 10 – Simulated vs Calculated Queue Size with A-Priori Traffic Stats

<u>Admission Parameters - Summary</u>

The ability to calculate queue size and delay from a-priori knowledge of the traffic statistics, versus knowing it with statistical confidence, is the difference between assured delivery on admission control decisions based on known delay bounds, and decision with some probabilistic confidence level. In the latter case, where the statistics are not assured, the more confidence is desired, the lower the efficiency of channel utilization will be. CMTS scheduling, and in particular HPRR scheduling [5], offer means to increase the confidence level of the statistical assurances, by supporting a best effort queue when the flow specification constraints are exceeded by a video stream. When servicing the excess video through a default queue, the relative delay will be impacted and be unpredictable, particularly if HSD services are added to the mix, adding stress to meeting the CPE delivery interval. However, the overflow queue provides an opportunity to successfully deliver packets that may otherwise be dropped when the statistics of the incoming streams cannot be assured.

We have quantified and simulated how maximum rate limiters, quanta, time-to-live counters [2], and (not shown here) minimum reserved rates can be combined with traffic characteristics to simulate network performance and result in quantifiable end-to-end network behavior. The multiple permutations of these relationships can be used to guide admission control decisions based on anticipating the impact of a new flow. While admission decisions have not been reduced it to a closed form expression – more of a multi-dimensional look-up table – the makings of the algorithm are as follows: calculate stats and existing workload, evaluate delay bounds/adjust, admit/deny/redirect source. The "redirect" step applies to the upcoming section introducing adaptive streaming into the model. Clearly, the more stream knowledge available a-priori or estimated directly the better, to the point of deterministic behavior for truly known statistics. Completely unknown inputs leave only mathematical characterization of how MPEG-4 AVC streams behave, as described in [2]. Unfortunately, this leaves a huge and impractical statistical range to accommodate.

Introducing Adaptive Streaming

The inclusion of adaptive technology as an emerging IP video tool promises new flexibility through a forgiving answer to the difficult yes/no admission problem, by serving up an answer that, instead of "No," can instead be "Yes, if…." We now take a closer look at how adaptive streaming technology influences IP video streams by incorporating a simple version into the model.

Consider a simple adaptive streaming model, where we are able to rate adjust the video, in this case using two different quantization levels. The basics of adaptive streaming, and in particular how it fits within the cable industry, are described in [3]. The stream multiplex is the same 18 HD VBR multiplex used in Table 6, using a four channel bonded DOCSIS 3.0 downstream. Note once again that the stream library used [4] is described in [2], and offers multiple format and encoding types to choose from. One particular video stream, a clip from *Transporter 2*, has a peak transmission rate of nearly 30 Mbps for high quality (high Peak Signal-to-Noise Ratio, or PSNR) as shown in Figure 11 (Q = 22), and higher still for even finer quantization.

Figure 12 shows the aggregate sent and received traffic with and without the adaptive mode turned on. Note how the initiation of the adaptive mode mitigates the peak burst that must be handled, reducing the "sent" volume. The resulting received traffic sequence now precisely follows the sent pattern through the now-reduced peak excursions. Note that a scale change of y-axis was used to expand on the tracking of the burst peak in the 250-265 sec range. A graphic artifact of the scale change is the ability to identify both sent and received flows on the "Adaptive" figure in red and blue, whereas on the wider, "No Adaptive" scale where there is overlap, the "received" traffic becomes hidden behind "sent" where they track.

.

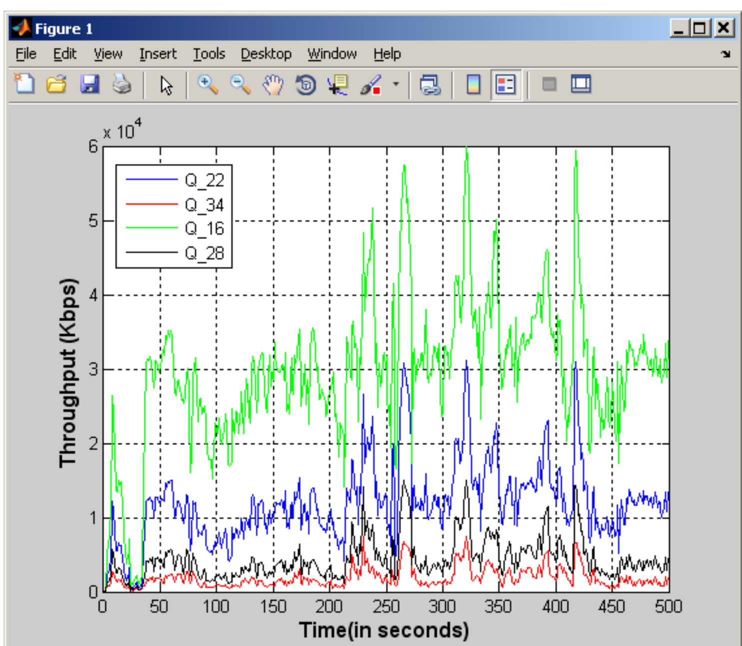


Figure 11 – Bit Rate vs QP for *Transporter 2* Clip

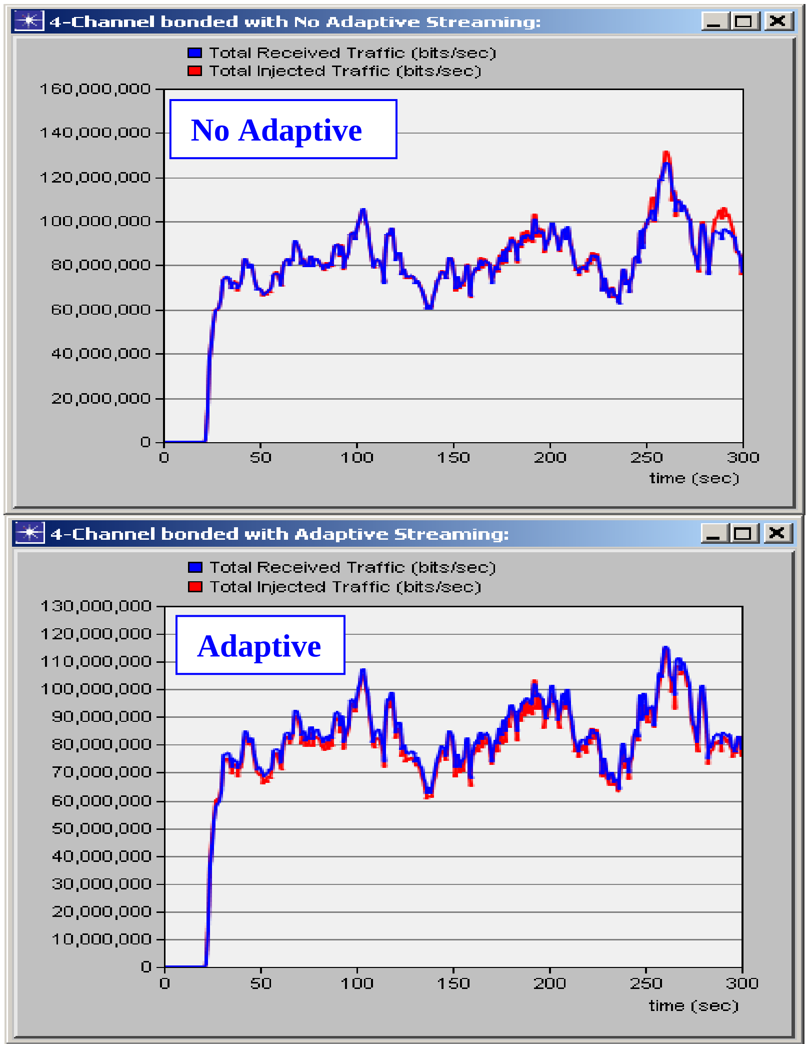


Figure 12 – Total Traffic Sent and Received: No Adaptive vs with Adaptive

The experience for the individual user watching *Transporter 2* is shown in Figure 13. (Editorial note – observing these bit rate plots of *Transporter 2* is actually more entertaining than watching *Transporter 2)*. The user suffers temporary picture loss during 250-265 and 282-297 sec time periods, identified by the top figure of Figure 13. We can estimate from Figure 11 that the latter period is likely not due to his or her movie, but instead likely due to peaks associated with others in the multiplex.

On the other hand, when adaptive streaming is turned on, the video server changes gears and sends a lower rate video clip than the primary stream. In this case, an encoding at Q_28 versus Q_22 is used. The resulting ability of the received traffic of the end user to follow the adaptive sent stream is shown in Figure 13, lower figure. The end-to-end packet delay is also lowered 22% compared to the non-adaptive case. The significance of this decrease is that it is another degree of freedom in system design – the trade-off of adaptive rates, or essentially transient video quality variation, in exchange for shorter buffers on the CPE side. In this example, a 400 msec buffer could have been reduced to nearly 300 msec. This can in turn translate to better user response for IPTV channel change.

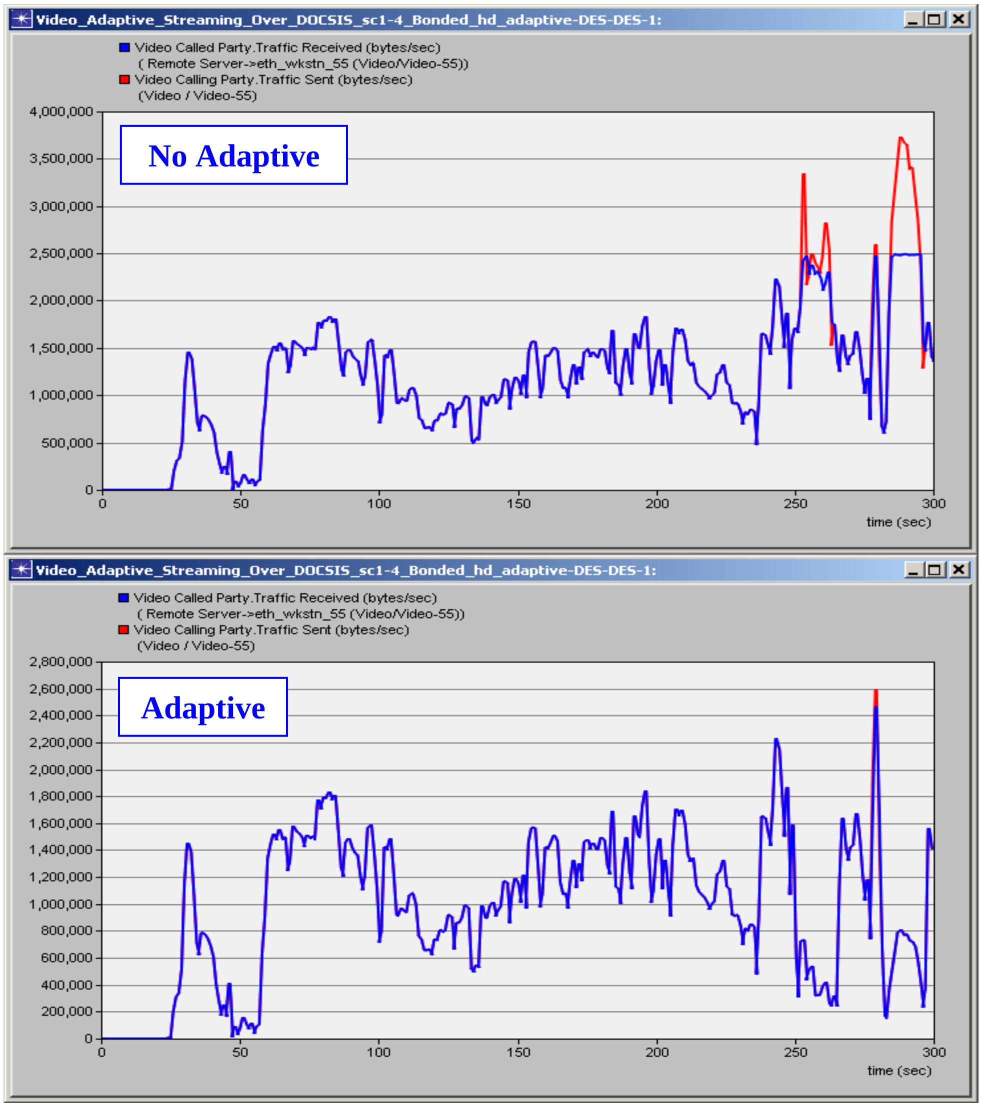


Figure 13: User Received Traffic: No Adaptive vs with Adaptive

VIDEO QoE WITH ERRORS

In the above simulations and the prior results referenced, different content types, formats, network variables, and technologies were permutated to understand the trade-offs involved in delivering low or no packet drop video service to IPTV users. Setting a drop threshold (1e-6), the assumption is that this threshold was chosen low enough to enable reasonable recovery mechanisms to handle clean-up. In the case of buffer size variation, 500 msec was assumed to be the maximum of what could be considered tolerable given system responsiveness needs. It should be noted that buffer sizes in terms of time translate to different memory sizes for different content types.

This section deals with the fallout of imperfect IPTV delivery when errors and drops ensue. We evaluate at the bit, byte, and packet level, where the latter would be the likely manifestation of congestion-oriented errors, and the former physical layer oriented. The byte error case can go either way, depending on other variables.

The current video delivery architecture, based on constant bit rate (CBR) streams encoded in MPEG-2, and using MPEG-2 TS over QAM transport, has some important operational advantages:

1) Simple traffic engineering and bandwidth management (CBR)
2) Low transmission errors (256-QAM)
3) Error resiliency (ITU J.83 encoded)
4) Assured timing/synch control (MPEG-2 TS)

For IP delivery, the advantage of a robust downstream physical layer remains. However, as has been discussed, bandwidth management aspects and timing assurances become more complex because of the use of VBR delivery, the dynamics of IP scheduling mechanisms designed for HSD, and the statistical probability of congestion that is not a component of existing video delivery.

To underscore the intolerance of video to transmission errors and packet drops, for which IP impairment mechanism would be randomized and potentially very harsh, a series of tests were performed whereby bit errors, byte errors, and transport packet loss was introduced into MPEG-2 and MPEG-4 video streams to observe how the displayed stream reacts. The impairments were introduced at steadily increasing rates and/or magnitudes. Subjective assessments were made by deliberately untrained eyes (not Video Quality (VQ) engineers) to better represent an average viewer experience. We do not proclaim equivalence to mean opinion score (MOS) levels of confidence, but the goal was to be more aligned with the home experience rather than the lab “find-the-irregularity” experience. A good lesson learned is never watch TV with a VQ engineer if you want to enjoy a program – they will find things that only Steve Austin (Google it, post baby-boomers) would otherwise identify.

Figure 14 shows a block diagram of the test setup.

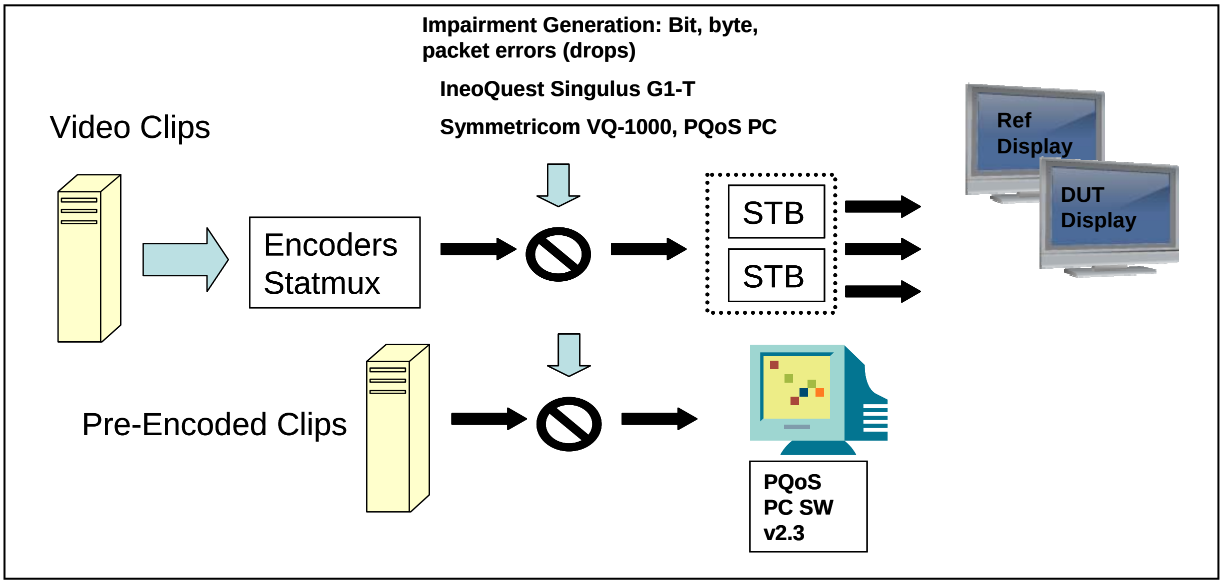


Figure 14 – Impairment Generation and Video QoE

Testing of digital video artifacts as a function of link quality and impairments exists throughout the technical literature. The results discussed here are not meant to recreate years of prior evaluations, but are primarily to provide a basis of observed actual content consistent with the mix and approach used in the simulations for comparison.

In addition, MPEG-4 part 10 encoding, while deployed in telco IPTV architectures, is still relatively early on the learning curve, and has evolved even since current deployments. Thus, any new insight observed adds to the growing library of experience with this standard.

Finally, some of the analysis tools, such as the latest revision of the Symmetricom PQoS video software analysis tool used, are also relatively new. Observations and results based on this tool thus offer potentially new data points in the continually evolving arena of subjective video quality analysis.

In addition to periodic packet dropping identified in Figure 14, implemented using the IneoQuest Singulus G1-T, the device also includes a test mode for rate reduction via dropping by hard peak capping. Though not described herein, this mode is an insightful complement to the testing described above, and represents the starkest possible contrast to the kind of intelligent rate control used by encoders, whose job is to maximize video quality at a particular bit rate allocation. Between the un-informed effects of IP video congestion delay, error, and bandwidth constraints, and VQ-based rate control, we have the ability to compare the best case and worst case ends of the impairment effect spectrum.

<u>Bit and Byte Errors</u>

Because of logistical constraints in the laboratory and VQ analysis tool, only MPEG-2 encoded content was available for the bit and byte error assessments. For MPEG-4 encoding, there are two obvious variables in play with respect to how it would compare, relatively speaking:

1) MPEG-4 is roughly one-half the bit rate on average, and higher in peak-to-average. Therefore, the same

periodicity of errors will effect twice as much, or nearly so, of the content from a time of occurrence and % of errors perspective

2) MPEG-4 has additional sophisticated filtering mechanisms design to reduce blur, halo, motion, and edge effects of block transform compression techniques. It is likely that these filters would act to positively impact potential artifacts (i.e. help to conceal them).

No further research (literary or test) was investigated as to whether, or under what conditions, these two factors cancel one another, or if one carries more weight.

Table 7 describes qualitatively the results of creating bit and byte errors for two types each of selected news-like and sports content using MPEG-2 encoded 720p HD.

For these streams, it is straightforward to observe from Table 7 that once bit or byte error rates stay below the 1e-6 range, viewing is unimpaired. Of course, bit transmission errors of this order and at least a couple orders of magnitude lower are generally well handled by FEC, particularly errors of the random type. When the effect of FEC is included, relatively graceful degradation such as observed in Table 7 gives way to perfect-or-objectionable, due to the nature of the FEC function. While FEC adds dBs of margin at a given error rate, it does so while steepening the error rate curve as a function of SNR.

For example, without FEC, a 1024-QAM downstream needs about 40 dB to achieve 1e-8 error rate. Referencing the "artifact-free" case from Table 7, it can achieve 1e-6 at about 38.2 dB, or about 2 dB lower. For an FEC applied that offered 3 dB of coding gain at 1e-6 (35.2 dB SNR required), we might find that the 1e-8 is achieved post-error correction at 35.7 dB, or a half dB different. The exact amount would vary by FEC architecture, but this represents the "steepness" effect. No specific architecture was called out here, because it is likely that when 1024-QAM arrives, there will be much discussion around deploying newer FEC structures than the now-dated ITU J.83 standard, such as newer low-density parity check codes (LDPC codes).

.

Table 7 – Bit & Byte QoE on MPEG-2 Encoded Content Types

	Content Type	
Bit Errors - One Per N Packets	"Easy" = News	"Hard" = Basketball
N = 100	Multiple simultaneous line and small block artifacts, constant	Multiple simultaneous line and small block artifacts, constant
N = 1000	Same as N = 100, just lesser in quantity, constant	Same as N = 100, just lesser in quantity, constant
N = 10,000	Same effect as N = 100,1000 but at roughly 5 sec intervals	Same effect as N = 100,1000 at ~5 sec intervals, less obvious (masked)
N = 100,000	One or two small block artifacts (mostly) per minute	None observed
N = 1,000,000	Nothing observed	Nothing observed
Byte Errors - One Per N Packets	"Easy" = News	"Hard" = Basketball
N = 100	Same effect as N = 100 "bits," aggravated occasional near break-up	Same effect as N = 100 "bits," aggravated occasional near break-up
N = 1000	Same effect as N = 1000 "bits," increase in block errors vs lines	Same effect as N = 1000 "bits"
N = 10,000	Same effect as N = 10,000, more often ~ every 2-3 seconds	Line artifacts observed every few seconds
N = 100,000	Line and small block artifacts every 15-30 seconds	One or two line artifacts per minute observed
N = 1,000,000	Nothing observed	Nothing observed

Byte errors stress the burst correction and interleaving elements of the receiver processing. Although in this testing we corrupted one byte at a time, multiple or consecutive byte error testing may be required to more finely define a threshold for impairments with this type of impact. Typically, error mechanisms are likely to be plant transient effects such as impulses of interference, power related spikes, or equipment malfunction anyplace there is electronics connected to the coax network. For the most part, byte errors acted like a worsened case of the same bit error rate, as might be expected. In the poorer cases of byte error frequency, another difference was that byte errors have a probability of resulting in a complete display break-up through overwhelming of the decoder's ability to make sense of the incoming information and effectively undo the encoding process.

An important phenomenon associated with the differences between objective measurement and perceptual experience is also apparent in the above table. That is, spot "popcorn" pixilation is more easily masked in some types of complex scenes. An "average viewer" would identify with three key elements on the screen that contribute to complexity:

1) Block-to-block detail granularity
2) High contrast sharp edges
3) Speed of motion

The first item above has the positive perceptual tendency of masking small macro-blocking when the brain is not expecting a pattern in the detail. In the sporting sense, the obvious example is crowd scenes, and even more so in panning crowd scenes as balls, pucks and athletes go past noise-like spectator backgrounds. As has been perceptually discovered over an over, people are wired to identify with patterns associated with prior experience, such as details in the action *on* the field, floor, or ice. Without a pattern to attract attention, fewer disturbances will be recognized. A second contributor to a better perception in this case is simply that the focus is on the match or game most of the time, not the spectators. This explains the better perception of more difficult content in the error injection tests in Table 7..

The part that suffers in the above example of high complexity is motion-related degradation. This tends to be associated, however, with constrained bit rate and infrequent packet loss, rather than the block and line pixelation associated with bit and byte errors. Note, however, that low motion scenes of great detail – also tested but not shown in Table 6 – where patterns *are* expected will get perceived differently than "noisy" detail. Examples are backdrops that involve high structure and high detail, such as cityscape or broad landscape scenes, or multitudes of faces at non-anonymous depth. In these cases, a perceptual expectation of detail is a prevailing factor.

Packet Drops – Effects & Recovery

Packet drops – MPEG or (worse) IP – show the intolerance of video delivery to packet loss. In doing so, it identifies the need for packet recovery mechanism when delivery cannot be deterministically assured, as is typically the case for IP data delivery.

Repeated Packet Drops

Table 8 describes qualitatively and quantitatively the packet dropping results of interrupting MPEG-4 AVC encoded Ethernet/IPv4/UDP transmissions of 720p HD content.

Table 8 – Packet Loss Impacts on MPEG-4 720p HD Streams

	Content Type	
Packet Drops - One Per N Packets	"Easy" = News	"Hard" = Basketball
N = 10	Freeze with no recovery	Freeze with no recovery
N = 100	Freeze with no recovery	SAME AS "EASY" CONTENT
N = 1000	Frames update every 1-2 sec	
N = 5,000	Freeze every ~5 sec with restart	Freeze every ~5 sec with restart
N = 10,000	Momentary freeze every ~7-8 sec	Momentary freeze every ~7-8 sec
One-Time Drop of N Packets	"Easy" = News	"Hard" = Basketball
N = 1	Macro-blocking and/or momentary freeze	Momentary freeze (no observed blocking)
N = 10	Momentary freeze	Momentary freeze
N = 100	Momentary freeze	Momentary freeze & sometimes artifacts on recovery
N = 1,000	1-2 Second freeze	1-2 Second freeze
N = 10,000	8-10 Second freeze	8-10 Second freeze

Repetitive (in this case with statistical regularity) packet loss creates frozen screens without recovery in the worst case, and periodic freezes of video in the best case – both clearly objectionable, in particular at the rates evaluated here. While unlikely, the repetitive case is valuable to observe in test because it ensures that we are statistically likely to encounter the effect of deleting an I-Frame. Loss of an I-frame certainly makes for more difficult recovery. In addition, while they represent a minority of the frames, I-frames could tend to be overrepresented as cases that cause congestion because of they are inherently larger than B and P frames.

Observing the impact given by the top-half of Table 8 gives a sense of the load that would need to be handled by an error mitigation mechanism, such as packet retransmission, due to link or routing related loss, going as low in this case of 1e-5 packet loss rate. Using a logical extrapolation from 1e-3 through 1e-5 effects, we would anticipate a momentary freeze on the order of a minute, give or take, for a 1e-6 case. Scaled by the user base served by an access device and/or servicing cache, this translates to some scale of processing load, memory, and signaling to manage for using packet recovery as part of a congestion management subsystem. This case (action every minute) would be the relevant relationship for an IP video system engineered based on the packet loss threshold defined in the simulations.

Single Burst of Packet Drops

The case of congestion based errors due to excess delay is more likely to lead to a series of packets being lost. Since behavior given a series of lost packets is important to understand, the bottom half of Table 8 shows the decoder recovery response when a one-time burst occurs that interrupts a packet sequence, varying in size from one to 10,000 packets. From 1-100 lost packets, the decoder recovery is roughly the same – instantaneous – following a momentary screen freeze. Nonetheless, this is an unacceptable experience, but also one likely every digital TV consumer has experienced. (Ironically, this happened to me just last evening watching from my DVR an excellent Episode 7 of the final season of "Lost." Unfortunately, the TiVo DVR in the middle muddies the water as to possible causes.)

Beyond the momentary freeze of relatively low loss events, momentary freezes become seconds of frozen screen when 1000 packets are dropped. Our simulations show that an unmanaged VBR congestion peak can create a sequence of drops on the order of 100's and in some cases 1000's. These larger sequences of lost packets, again multiplied by the scale of the user base accessing the recovery mechanism, are again indicative of the order of memory, control, and network bandwidth necessary to create effective packet recovery.

There are similar, albeit less apparent, content based differences as a function of the size of the dropped sequence. For a small enough sequence of packet drops, the less likely case for network related condition, we see in Table 8 again that the complexity level contributes to some masking of the blocking effects of the decoder upon recovery. In contrast, as the burst event size increased to N=100, the complexity of the scene – in particular the motion complexity – created added stress on the decoder to recover with the fidelity of the low-complexity news content.

Video Rate Reduction

As previously described, a basic principle of video encoding is to maximize the quality for a given allowable bit rate of delivery. The science and literature is rich with objective and subjective analysis of the multiple variables in play to achieving this, and encoder manufacturers spend time and effort developing solutions that continue to optimize these relationships. In all cases, it is a basic premise to deliver optimal perceived quality (a subject all to its own) at minimum bit rate, which results in VBR streams at a given pre-determined quality.

Now, with the information available to encoders as part of the compression algorithm process – real-time knowledge of scene complexity – the reverse relationship can also be explored. That is, rather than starting with defining a desired level of quality, a given available bit rate can be the independent variable, from which an optimum perceived output can be derived. This basic premise is at the heart of adaptive streaming protocols aimed at improving the experience of Internet video – and more broadly video over IP in general. While our network simulations implement hard capping (i.e. network simulation tools pay attention to link and packet level performance, not video quality) capped VBR conclusions drawn in the simulations to ensure network performance would be applied with awareness just as in the adaptive case, as long as we can assure that a reasonable video quality is achievable within the capping limit defined. In the network modeling world, this correlates to ensuring a cap that is a reasonable rate reduction as a percentage of the peak.

CONCLUSION

Delivering video over DOCSIS, with the same QoE or better, as the existing MPEG-2 TS based infrastructure is critical for a successful transition of services. However, IP traffic has historically always been limited in its ability to provide deterministic delivery guarantees. Data and voice services, to an extent, are robust to some subset of the potential obstacles. Video is robust to none of them. There are some positives compared to voice service – the most notable exceptions being a pure latency advantage (less sensitive) and some added flexibility in dealing with jitter. These come at orders of magnitude differences in service processing bandwidth, however.

We have looked at an array of variables associated with IPTV delivery as the traffic engineering of this architecture begins to take shape. In particular, we observed once again the strong content dependence of video streams on the bandwidth efficiencies.

In addition to the external variables such as CPE buffer size, packet error metrics, and peak-rate capping, we took advantage of some predictability in a video-only service class configuration to see how making use of some of the basics of video packets translates into better network performance. We added adaptive capability to the simulation and observed how this enabled dynamic traffic injection to be more ably followed, packet-for-packet, at the CPE, with no information loss. Finally, we recognized that network simulations gathering numbers do not capture perceptual video effects. Thus, using a similar content mix basis, we took a look at what error and loss mechanisms mean to end user QoE, and how packet loss translates to display and recovery times. This information estimates the scale of the problem for a suitable packet recovery mechanism as a function of a range of packet loss rates and sizes.

Most importantly, we have further developed a very robust model that can be used to comprehensively understand all aspects of DOCSIS delivery of video services, and additionally can be used to evaluate performance for converged services over bonded DOCSIS 3.0 downstreams.

REFERENCES

[1] Dr. Robert Howald, "Fueling the Coaxial Last Mile,"2009 Society for Cable Telecommunications Engineers (SCTE) Emerging Technologies Conference, Washington, DC, April 3, 2009.

[2] Dr. Robert Howald, "Web Surfing to Channel Surfing: Engineering the HSD Edge for Video," 2009 Cable-Tec Expo, sponsored by the Society for Cable Telecommunications Engineers (SCTE), Denver, CO, Oct 28-30, 2009.

[3] John Ulm, Tom du Breuil, Gary Hughes, and Dr. Sean McCarthy, "Adaptive Streaming – New Approaches for Cable IP Video Delivery," Cable Show Spring Forum, May 11-13 2010, Los Angeles, CA.

[4] Fitzek, Frank H.P. and M. Reisslein, "MPEG-4 and H.263 Video Traces for Network Performance Evaluation," (extended version), Technical Report TKN-00-06, Technical University Berlin, Dept. of Electrical Eng., Germany, October 2000.

[5] Patrick, M., "Hierarchical prioritized round robin (HPRR) scheduling," US Patent No 7457313, August 2005.

ACKNOWLEDGMENTS

The authors would like to thank the following individuals for their insights and support in developing this paper: Sean McCarthy, Stan Egger, Xiahan Zhu, and Howie Ton-That, John Ulm, Erik Metz, and Mike Patrick.