Throughput Analysis of a Broadband Multimedia Satellite Fast-Packet Switch

Jaime L. Prieto, Jr., Scott L. Wilkinson and Jason R. Kincaid
TRW Space & Electronics Group / Network Systems Engineering
One Space Park, Redondo Beach, CA, USA 90278
***Business*: joe.freitag@trw.com, *Networks*: jaime.prieto@trw.com, *Comm*: eric.wiswell@trw.com**
www.trw.com

ABSTRACT

Ka band multimedia satellite networks have the potential to bring two-way broadband services directly to the end-user while providing ubiquitous global connectivity with Quality-of-Service (QoS) guarantees. TRW's satellite network architecture design process involves *Demand*, *Design* and *Financial* modeling. The main *Design* challenge is to balance the delivery of network services and QoS demanded by users, minimize the size, weight and power (SWAP) of the processing satellite, maximize the throughput-revenue capacity, and minimize the overall cost of the network. A basic design tradeoff exists between the throughput of the satellite switch and the QoS delivered to the end-user. Mechanisms to ensure QoS in a satellite are constrained by the satellite SWAP. An important component of the satellite SWAP is the amount of memory needed to buffer data packets, which is directly related to the throughput and Cell Loss Ratio (CLR) achievable in the fast-packet switch.

This paper describes a small sample of some of our early simulation and analysis efforts to start the architecture design cycle, where the focus was on the throughput of the switch and output port buffer requirements. The analysis assumes geographically dispersed users running multimedia applications with two-way broadband connectivity through the satellite. The paper further illustrates how analysis and simulation contribute to a successful network design.

INTRODUCTION

In early 1997, TRW's network systems engineering team began to explore the requirements for satellite-based fast-packet switching. Early network architecture design work was based on the hypothesis that higher returns on investment would be generated by the increased throughput of the fast-packet switch compared to legacy satellite systems. Since then, significant debate has taken place within the satellite community regarding the value of a switching "processed" satellite system. TRW quantified the debate by systems engineering and business analysis, documented in what we call the "value paper".[1]

This paper describes the analysis and simulation results used to estimate throughput and buffer sizing requirements of the satellite fast-packet switch[*] in support of the "value paper." At the time that this study took place, simulation results were not sufficient due to the long simulation times required to estimate Cell Loss Ratio (CLR) with a high level of confidence. Nevertheless, simulation was used to characterize the behavior of different types of traffic on an idealized switch design (i.e. as a complimentary tool to analysis). We start this paper with a description of our traffic demand models, followed by a description of the ideal switch design models used. The third section describes the equivalent bandwidth analysis of multimedia traffic and Call Admission Control (CAC). The final section describes the simulation of homogeneous and heterogeneous applications traffic.

TRAFFIC DEMAND MODELS

User Application Source Models

Users, the source of demand, come in three general categories, such as service providers, business users, and residential users. The analysis presented in this paper only considered the residential users, where there is one user per terminal (UT) per call. The applications considered in the analysis are voice telephony, video-conferencing, broadcast-quality video and Web surfing; each user executes one application at a time. The uplink access model is based on the uplink rates available in the TRW Gen*Star link-layer design. For the voice and video real-time applications, the smallest uplink rate that met the source peak rate was assigned as dedicated bandwidth. No further smoothing was performed for the source application models. We selected the four representative applications shown in Table 1 for analysis. The web surfing model assumed exponential message lengths, with the average web-request = 320 Bytes, and the average reply size[†] is 10 KBytes. The user think-time had a median of 16 seconds, giving a combined average rate of 5.28 Kbps[2]. The web surfing application was assigned shared bandwidth through a demand-assigned

[*] In this paper, a packet or "fast-packet" is used interchangeably with an ATM cell.

[†] We have since modified our web models to incorporate "standard" empirically based models from SPEC, available at www.specbench.org/osg/web96/webpaper.html

multiple access (DAMA) scheme with an efficiency of 50%, selected as a conservative estimate[‡]. The first of two video sources is broadcast-quality MPEG Variable Bit Rate (VBR) video encoded for video-streaming. A collection of actual MPEG sequences scaled to a target average rate were used for the simulation portion of the analysis. A Constant Bit Rate (CBR) video source model based on ITU-T H.261 was selected for videoconferencing at 384 Kbps. The voice stream was encoded at 16 Kbps using ITU-T G.728 with a matching silence suppression algorithm.

Table 1. Application Source Model Description

User Application	***Average Rate per Connect (Kbps)***	***Peak Rate per Connect (Kbps)***	***Peak-to-Average Ratio (unitless)***
Web Surfing	Client =0.16 Server=5.12	Client = 768 Server=3,072	Client = 4,800 Server = 600
Broadcast Quality Video	4,000	8,000 (smoothed)	2.0
Video Conference	384	384	1.0
Voice	6.4	16	2.5

Application Traffic Scenarios

TRW bounded the trade space for each application. Operational traffic is assumed to fall between these extremes. Table 2 shows the selected application traffic scenarios as a percent of downlink bandwidth.

Table 2. Application Traffic Scenarios % Bandwidth

User Apps.	***1. Equal Distribution %***	***2. Emphasis on BCast Video %***	***3. Emphasis on Web %***	***4. Emphasis on Voice & VideoConf %***
Web Surfing	25	20	80	34
BCast Video	25	50	6	6
Video Conf.	25	17	7	30
Voice	25	13	7	30

IDEAL FAST-PACKET SWITCH MODEL

The purpose of these initial tests was to bound the throughput performance of the switch output buffers based on the buffer occupancy of rate-limited source application models. For an upper-bound on the output buffer occupancy, an infinitely fast non-blocking switching unit was assumed. The detailed performance of the switching unit was the subject of a different study, and is not addressed in this paper. Therefore, the idealized model used for this analysis can be described as a G/D/1 queue (general source arrival distribution, deterministic fixed packet sizes, and one server). The analysis presented assumes 128 * 1024 = 131,072 cell output buffers, whereas the simulations used infinite buffers in order to measure actual occupancy. The satellite uplink and downlink beams (i.e. switch input and output ports) are at the rate of 155.52 Mbps, or 366,792 cells per second using direct cell transfer without physical and link-layer overhead. The downlink scheduler used for this study was the straight non-preemptive priority service discipline, where video-conferencing had the highest priority, followed by voice, and broadcast quality video. The web surfing packets had the lowest priority. It should be noted that system parameters used here may or may not represent the actual TRW product.

EQUIVALENT BANDWIDTH ANALYSIS OF MULTIMEDIA TRAFFIC

This section presents the throughput analysis performed on the idealized satellite fast-packet switch given the user application source models defined in earlier sections. Equivalent bandwidth (EB) is the effective statistical amount of bandwidth required by a connection to maintain QoS guarantees through a shared bandwidth $\boldsymbol{C_{link}}$ and a buffer size $\boldsymbol{B}$. The EB presented uses *On-Off* models, as shown in Figure 1. The *On-Off* model captures important traffic dynamics and is simple enough to be useful[3] for CAC. While *Off*, no traffic is generated, and the *Off* time is exponential with average length θ_u^{-1} seconds. While *On*, traffic is generated at a constant peak rate $\boldsymbol{R_u}$, and the *On* time is exponential with average length β_u^{-1} seconds. The source activity, or probability of being *On* is $\boldsymbol{a_u} = \theta_u / (\theta_u + \beta_u)$. Similarly, the average burst size is $\boldsymbol{b_u} = \boldsymbol{R_u} / \beta_u$ in cells. Each user connection $\boldsymbol{u}$ is characterized by the average arrival rate $\lambda_u = \boldsymbol{a_u} \boldsymbol{R_u}$ (*cells/second*) and the variance of the arrival rate $\sigma_u^2 = \boldsymbol{a_u}(1-\boldsymbol{a_u})\boldsymbol{R_u}^2$ (*cells/second*)^2.

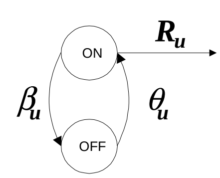

Figure 1. *On-Off* Model

Assuming that the number of sources $\boldsymbol{N}$ is large enough that the Central Limit Theorem applies, then the total traffic

‡ The 50% efficiency factor of the DAMA scheme was determined by other analysis, and has the effect of requiring twice the average rate. DAMA efficiency is highly dependent on the actual source traffic and uplink scheduling algorithm.

behaves as a *Gaussian* process with the following mean and variance of the arrival rate:

$$\lambda = \sum_{u=1}^{N} \lambda_u ; \sigma^2 = \sum_{u=1}^{N} \sigma_u^2$$

Equation 1. Mean and Variance of the Arrival Rate

The burst-coefficient c_u^2 is shown below as a function of the *On-Off* model parameters as

$$c_u^2 = \frac{Var[IAT]}{E[IAT]^2} = \frac{1-(1-\frac{\beta_u}{R_u})^2}{(\frac{\beta_u}{R_u}+\frac{\theta_u}{R_u})^2} = \frac{1-(1-b_u^{-1})^2}{(b_u^{-1}+\frac{\theta_u}{R_u})^2}$$

Equation 2. Squared coefficient of variation of the cell interarrival times

Finally, $\alpha_u = \lambda_u c_u^2$ is the per-connection cell arrival process instantaneous variance is given by

$$\alpha = \sum_{u=1}^{N} \lambda_u c_u^2 = \sum_{u=1}^{N} \alpha_u$$

Equation 3. Cell arrival process instantaneous variance

A summary of all *On-Off* model parameters is shown in Table 3. It should be noted that all user applications are modeled well by the *On-Off* process, except for the broadcast video application. The broadcast video *On-Off* parameters were estimated using a technique by Guerin[4] based on measurements[§].

Table 3. User Application *On-Off* Model Parameters

	R_u	β_u	θ_u	a_u	λ_u	σ_u^2	c_u^2
Application	*cells/sec*	*seconds* $^{-1}$	*seconds* $^{-1}$	*ON prob*	*cells/sec*	*(cells/sec)^2*	*unitless*
Voice	41.67	2.50	1.67	0.40	16.67	416.7E+0	11.6E+0
Broadcast Video	20833.33	4.42	11.93	0.73	10416.67	85.6E+6	689.0E+0
VideoConference	1024.59	20.49	833.33	0.98	1000.00	24.6E+3	57.0E-3
Web Client	2000.00	300.00	0.06	208E-6	0.42	830.1E+0	12.3E+0
Web Server	8000.00	37.50	0.06	1.7E-3	13.33	106.3E+3	424.2E+0

Diffusion-based Equivalent Bandwidth

Diffusion-based equivalent bandwidth and associated CAC algorithms seem to generate results that are consistent with observed performance. The resulting equations are computationally efficient, and are a function of the buffer size, the target cell loss ratio (CLR), and the *On-Off* parameters of the active application models. The form of the diffusion based equivalent bandwidth[5] chosen for analysis is the finite buffer cell loss estimation and is given by Equation 4.

$$C_{df} = \lambda - \delta + \sqrt{\delta^2 - 2\sigma^2\omega}$$

$$where \quad \delta = \frac{2B}{\alpha}\sigma^2, \quad \omega = \ln(CLR\sqrt{2\pi})$$

Equation 4. Capacity using diffusion-based equivalent bandwidth

The units of C_{df} are the same as λ, in cells/second. A new value of C_{df} is calculated when a connection is added, modified or deleted from a link. The usage of the equivalent bandwidth in relation to the link capacity is the subject of the next section on CAC.

Call Admission Control (CAC) Algorithms

CAC is the process of determining if a new call is to be admitted to the network without violating traffic contract QoS requirements of the new and existing users. For a review of other CAC algorithms, see reference [6]. The CAC procedure presented here is based solely on meeting a CLR requirement, and does not cover other important metrics, such as bandwidth fairness, delay, and delay variation. Nevertheless, the algorithm presented was deemed sufficient for our early estimates and it provides a good foundation for a more accurate and complex CAC. The procedure for CAC is a simple three-step process:

1. Maintain the information vector $I = \{\lambda, \sigma^2, \alpha\}$ at each switching node in the network, in addition to the total buffer size B and the overall CLR requirement[**]
2. Calculate C_{df} for the existing calls plus the new call seeking admission
3. If the new $C_{df} < C_{link}$, then ACCEPT the call; REJECT otherwise.

In order for the new call to be admitted to the network, the bandwidth needed to support it must be available at each and every link in the route from source to destination. Figure 2 shows the CAC processes that occur in attempting to establish a call from source to destination users.

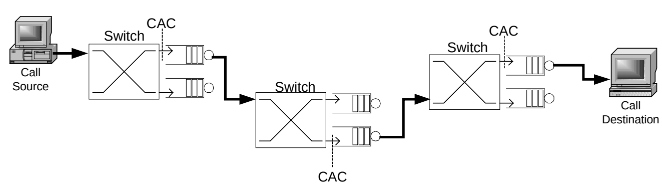

Figure 2. Call Admission Control processing

[§] Parameters are chosen such that the impact of the complicated input process on bandwidth requirements is 'essentially' captured by its exponential approximation. More work needs to be done in the area of modeling and bandwidth estimation of VBR video sources; the parameters listed represent a first-iteration only.

[**] The CAC algorithm presented makes the unfortunate assumption that the applications traffic is homogeneous and has the same CLR requirement.

Throughput Analysis Results

We proceed to estimate the maximum throughput of an ideal switch with an output buffer size of $\boldsymbol{B}$ = 131,072 cells and a satellite downlink rate of $\boldsymbol{C}_{link}$ = 366,792 cells/second for CLRs: $\{10^{-6}, 10^{-7}, 10^{-8}, 10^{-9}\}$. The link load is the fraction of the link capacity with actual traffic, and is defined to be ρ = *Throughput* / $\boldsymbol{C}_{link}$. The number of connections in a downlink beam is given by *BeamConnections* = *floor*(*TargetRate* $\ast$ *ApplicMix* / *AverageRate*), where *TargetRate* = $\rho \ast \boldsymbol{C}_{link}$ and the actual traffic through the shared link bandwidth is *Throughput* = *BeamConnections* $\ast$ *AverageRate*. *BeamConnections* is a vector with the number of connections for each application, *ApplicMix* is the percent downlink bandwidth of each application from Table 2, and *AverageRate* is a vector with the average rate from Table 1 for each application. The "*floor*()" function gives the smallest integer resulting from the calculation (partial connections are not allowed). The "$\ast$" function performs an element by element multiplication of the vectors. Additional web surfing connections were added after the initial allocation in order to get closer to the *TargetRate*, as *Connections*(web) = *Connections*(web) + *floor*((*TargetRate* – *sum*(*Throughput*)) / *AverageRate*(web)), where *sum*() adds up the throughputs of all application elements of the vector. In each case, the maximum load ρ_{max} was calculated using the equivalent bandwidth equations presented earlier for the given target user applications mixture using an iterative root-finding procedure such that all active connections met the CLR requirement under test. Stated mathematically, this becomes $\rho_{max} = 1 - \mathrm{MAX}(\boldsymbol{C}_{df}) / \boldsymbol{C}_{link}$, such that $0 < \rho_{max} < 1$.

Homogeneous Traffic

This section presents the throughput analysis results obtained by considering the queuing behavior of individual application sources (i.e. homogeneous traffic) on the G/D/1 queue. Figure 3 shows the experimental setup for the videoconferencing application (without the others).

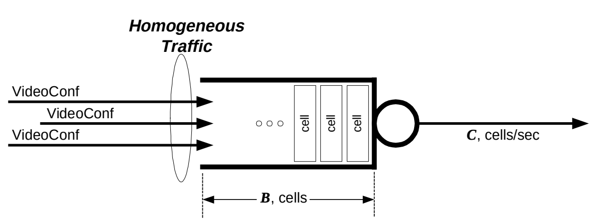

Figure 3. Homogeneous Traffic into a G/D/1 Queue

Figure 4 shows the resulting maximum load for each user application. A loading of 1.0 is predicted by the CAC algorithm for voice and videoconferencing applications for all CLRs; this is expected given the low burst-coefficient of the application sources (i.e. ~ smooth traffic) and the extremely large buffer size available in the output port. The drop in throughput for web and broadcast-quality video is the result of the significant increase in burst-coefficient (i.e. bursty traffic) relative to the smooth applications. The maximum loading predicted for web surfing ranges from 0.97 to 0.98, whereas broadcast quality video ranges from 0.95 to 0.97.

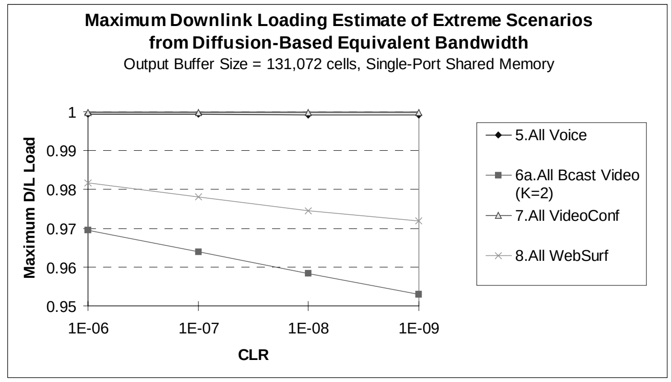

Figure 4. CAC results using homogeneous traffic

Again, the burst-coefficient predicts higher throughputs for web surfing compared to broadcast video. The broadcast quality video CAC is shown for a peak-to-average ratio of $\boldsymbol{K}$ = 2. A smaller $\boldsymbol{K}$ indicates more smoothing[7] on the sources. Selecting the value of $\boldsymbol{K}$ is a tradeoff between throughput and delay; the more smoothing, the greater the delay and the throughput of the individual streams through a single node.

Heterogeneous Traffic

This section presents the throughput analysis results obtained by considering mixtures of application sources (i.e. heterogeneous traffic) on the G/D/1 queue. Figure 5 shows the experimental setup, where application mixture scenarios are fed into the queue.

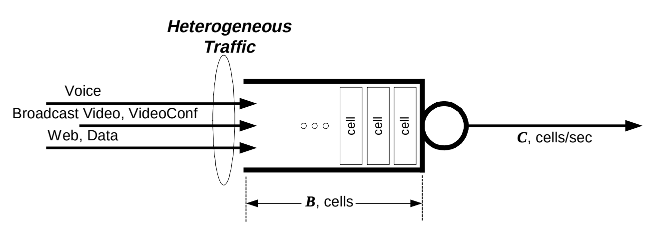

Figure 5. Heterogeneous Traffic into a G/D/1 queue

Figure 6 shows the predicted maximum load for user application mixtures (non-homogeneous traffic), where the aggregate burst-coefficient becomes a function of the relative amounts of smooth and bursty sources. As expected, the applications mixture scenario with the highest throughput is scenario 4, which emphasizes voice and videoconferencing (~ smooth sources); they achieved loads of about 0.99. On the other extreme, scenario 2, which emphasizes the broadcast-quality VBR video, had the overall lowest throughput ranging between 0.97 and 0.98 due to the large burst-coefficient of MPEG video.

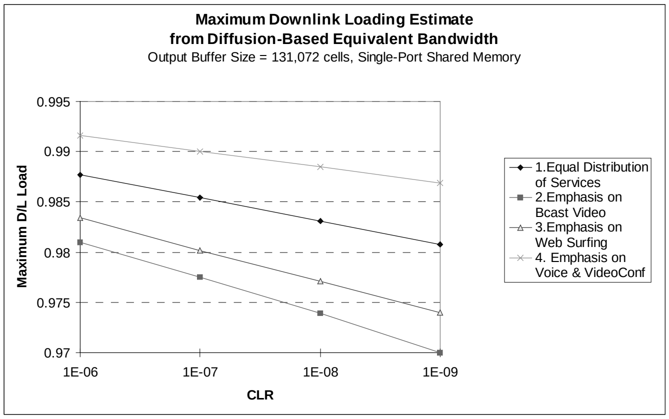

Figure 6. CAC results using non-homogeneous traffic

SIMULATION RESULTS

This section presents the simulation results used in the throughput analysis of an ideal satellite fast-packet switch.

Homogeneous Traffic

This section presents the simulation results of individual user applications traffic (homogeneous) on the buffer occupancy of a single server queue. Figure 7 shows the cumulative mass function (CMF) of the buffer occupancy for videoconferencing (on the left) and voice (on the right). The buffer occupancy CMF for videoconferencing, which has a burst-coefficient << 1.0, clearly shows that smooth CBR-like applications do not require much buffering even at loads approaching 100%. The buffer occupancy never reached 100 cells during the simulation. The silence suppressed voice, which has a burst-coefficient ~ 12, starts to show buffer occupancies reaching 1,000 cells.

Figure 8 clearly shows the effect of sources with high burst-coefficients. The buffer occupancies for both broadcast-quality video and web surfing are 1 to 2 orders of magnitude larger than the smooth traffic sources. In the worst-case, the broadcast-quality video required buffering up to 100,000 cells at the 0.98 loading (i.e. the highest loading considered).

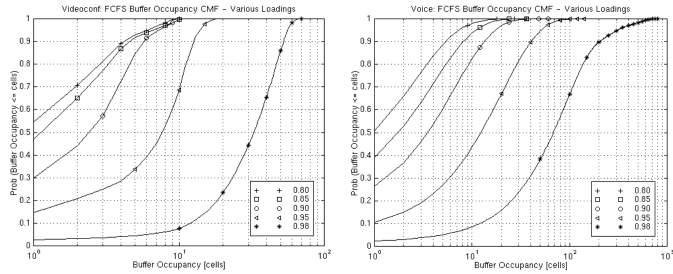

Figure 7. Smooth Applications Traffic Buffer Occupancy Cumulative Mass Function

The web surfing application, on the other hand, required buffering of roughly 40,000 cells. It should be noted that these buffer size estimates are based on the maximum buffer-occupancy observed for a 45 second simulation. Much longer simulation times or a combination of analysis and simulation are required for accurate buffer sizing. The simulated times are short because these scenarios take a lot of computing time to execute. For instance, accurate buffer sizing for web surfing at 0.96 load with a CLR requirement of 10^{-9}, would take roughly 1,369 days (3.75 years) to run. Therefore, other techniques, such as importance-sampling[8] are needed to increase the statistical confidence in the buffer size estimates.

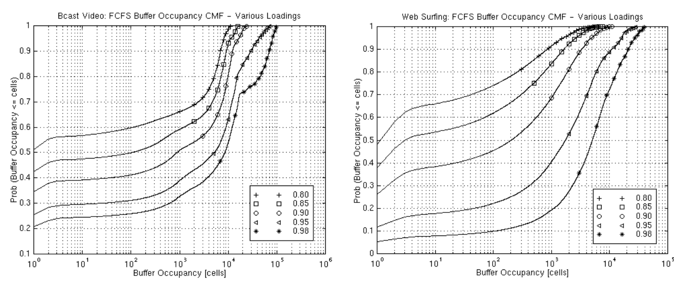

Figure 8. Bursty Applications Traffic Buffer Occupancy Cumulative Mass Function

Figure 9 shows the effects of the individual applications traffic types at the same load of 0.98. As expected, videoconferencing has the smallest buffer occupancy. Videoconferencing is closely followed by the Poisson packet arrivals. It is interesting to note that the Poisson assumption seems to provide an upper bound for CBR traffic[††], but underestimates buffer requirements by several orders of magnitude for VBR traffic. Web surfing and broadcast video are shown to require much larger buffers than the smooth applications.

[††] The buffering, and thus variation of CBR traffic is due to the random start times of the periodic applications streams.

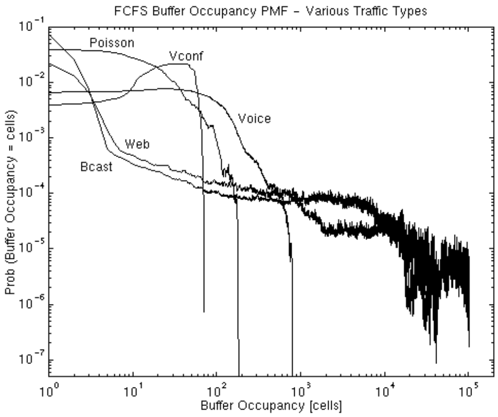

Figure 9. Comparison of User Applications Traffic Buffer Occupancy Histograms at 0.98 Load

Heterogeneous Traffic

This section considers user application mixtures, where connections of the same application are aggregated into the same buffer. Buffers are serviced using a non-preemptive priority discipline, where the higher priority buffer always gets service provided it has data packets to send. The highest priority is assigned to videoconferencing, whereas the second priority is given to voice. Third and fourth priorities are assigned to broadcast video and web surfing respectively. Figure 10 shows the buffer occupancy of each application's buffer for the applications mixture scenario emphasizing it. For instance, the voice buffer occupancy is shown for the applications mixture scenario 4 emphasizing voice; the buffers needed here are relatively small – less than 10 cells. On the other hand, the buffers needed for the bursty traffic are 3,000 and 40,000 cells, for web and broadcast video, respectively. The results presented in this section emphasize the effects of burstiness on the required buffer sizes needed to meet a CLR requirement, without considering the time variations of the buffer occupancy. For a given memory pool, shared memory between output ports may be used for additional gains in throughput or a reduction in buffer space required.

CONCLUSION

Models for the multimedia applications of silence-suppressed voice, two types of video, and web surfing were presented. An algorithm for call admission control (CAC) using diffusion-based equivalent bandwidth was used to estimate maximum throughputs for homogeneous and heterogeneous traffic. Given a link capacity of 155.52 Mbps, a buffer of 128 K cells, and a CLR of 10^{-8} (without considering delay and delay variations in the throughput analysis), we estimate a maximum throughput of 0.98 for the ideal fast-packet switch loaded using an equal distribution of applications mixture scenario.

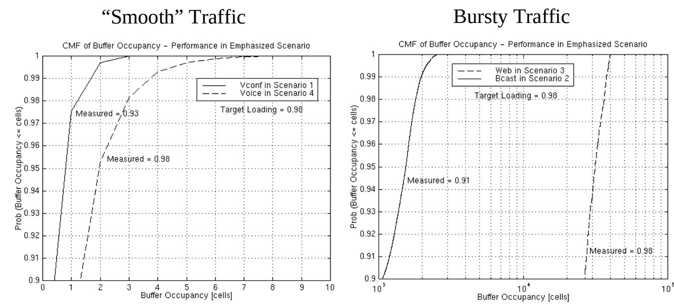

Figure 10. Effects of User Application Mixtures on Buffer Occupancy

Future work includes the effects on throughput and QoS of smoothing, policing, shared memory, early and partial packet discard, TCP/IP, weighted fair queuing, ABR, GFR, per-connection queuing and multicast traffic over a satellite network. We have shown that both analysis and simulation are needed in the network architecture design process.

REFERENCES

[1] Mark Bever, Joseph Freitag, Stuart Linsky, James M. Myers, Raymond M. Nuber, Jaime L. Prieto, Jr. and Eric R. Wiswell, "Fast-Packet vs. Circuit Switch and Bent Pipe Satellite Network Architectures," Fourth Ka-Band Utilization Conference, Venice, Italy, Nov. 2- 4, 1998

[2] Bruce A. Mah, "An Empirical Model of HTTP Network Traffic," IEEE Infocom '97, Kobe, Japan, April 1997, at http://www.ca.sandia.gov/~bmah/Software/HttpModel/

[3] Mischa Schwartz, "Broadband Integrated Networks," Prentice Hall, 1996.

[4] Roch Guerin and Levent Gun, "A Unified Approach to Bandwidth Allocation and Access Control in Fast Packet-Switched Networks," IEEE Infocom '92, pp. 1 – 11.

[5] Erol Gelenbe, Xiaowen Mang, and Raif Onvural, "Bandwidth Allocation and Call Admission Control in High-Speed Networks," IEEE Communications Magazine, May 1997, pp. 122 – 129.

[6] Harry G. Perros, and Khaled M. Elsayed, "Call Admission Control Schemes: A Review," IEEE Communications Magazine, Nov. 1996, pp. 82 - 91

[7] Steven Gringeri, et. all, "Traffic Shaping, Bandwidth Allocation, and Quality Assessment for MPEG Video Distribution over Broadband Networks," IEEE Network, November/December 1998, pp. 94 – 107.

[8] J. Keith Townsend, Zsolt Hraszti, James A. Freebersyser, Michael Devetsikiotis, "Simulation of Rare Events in Communications Networks," IEEE Communications Magazine, August 1998, pp. 36 – 41.