Technical Report 2003-02-04
Internetworking and Media Communications Research Laboratories
Department of Computer Science, Kent State University
http://medianet.kent.edu/technicalreports.html

Performance Analysis of TCP Interactive based World Wide Video Streaming over ABone: Symbiotic Control and Video Quality

Javed I. Khan and Raid Y. Zaghal
Networking and Media Communications Research Laboratories
Department of Computer Science, Kent State University
233 MSB, Kent, OH 44242
javed|rzaghal@cs.kent.edu

Abstract—Interactivity in transport protocol can greatly benefit transport friendly applications. We have recently implemented an interactive version of TCP. The implementation has two components-- an interactive transport protocol over FreeBSD called iTCP and, a novel symbiotic MPEG-2 full logic transcoder, which can dynamically change video characteristics based on interactive congestion response inside network layer. We have experimented with the real system on the Active Network (ABone) using selected nodes in the U.S. and Europe. In this report we present the application level improvement in video spatial and temporal quality experiments of the live video streaming results to these sites. A second report contains the detail results from network level congestion and delay/ jitter experiments.

1. Introduction

This is the second technical report of our experiment on the ABone. In the first report [KhZ03R2] we discussed jitter and delay management of the video stream. In this report we present event trace and symbiotic rate control. We also focus on video quality parameters and quality-delay tradeoff.

The complete introduction, related work, and background work can be found in our first report [KhZ03R2].

2. The ABone Active Networks

An important feature of our experiment is using a real implementation of the proposed transport protocol (iTCP) and the MPEG-2 transcoder. Furthermore, we wanted to run the experiment on the real Internet environment. To conduct our experiment we wanted to run our video player on a number of remote hosts around the world and measure performance in each case. We could have done this by "*telneting*" to those remote nodes. But this would have required preparation and communication with people around the world to setup accounts and administer them. Furthermore, this will not be flexible nor practical if we decide to switch to a new set of remote nodes. Therefore, we decided to use the ABone Active Networks.

In Active networks, the routers or switches of the network can perform customized computations on the messages flowing through them. These networks are active in the sense that nodes can perform computations on the contents of the packet. As far as we are concerned, we wanted to be able to run our video player on a selected set of ABone nodes and measure the performance of the video session. In that respect, the ABone provided a convenient testbed for us to run the experiment. We simply sent a modified version of our video player to the ABone administrator at the ABone Coordination Center (ABOCC) to be placed on the trusted code server at (http://bro.isi.edu/KENT). Then we configured and registered our iTCP machine (kawai.medianet.kent.edu) as a primary node on the ABone.

In addition to the iTCP machine we have 10 registered ABone nodes at Kent State University (mk00- mk09.maunakea.medianet.kent.edu). Four of these nodes run on FreeBSD and the rest run on Linux. At the time of our experiment (Feb. 2003) there were 24 Linux nodes, 5 Solaris nodes, and 12 FreeBSD nodes registered on the ABone. Since our player was compiled on Linux, we could use Linux nodes only.

3. Experiment Setup

ABone node	IP	Country	Number Of Hubs	RTT (ms)		
				Avg	min	max
abone.fokus.gmd.de	193.175.135.49	Denmark	21	144	131	216
galileo.cere.pa.cnr.it	147.163.3.12	Italy	20	287	266	339
abone7.cs.columbia.edu	128.59.22.217	NY, USA	15	41	39	60
abone-01.cs.princeton.edu	128.112.152.62	NJ, USA	15	51	46	69
dad.isi.edu	128.9.160.202	CA, USA	16	65	65	68

Table-1. ABone nodes used to run the player in the experiment.

This experiment describes the performance for the case of a MPEG-2 ISO/IEC13818-2 (176x120) resolution video encoded with base frame rate of 2 Mbps at main profile on the symbiotic transcoder. Figure-1 illustrates the experiment setup. The video server runs on a classic TCP machine (*manoa*) and feeds the video stream into the transcoder, which runs on the iTCP machine (*kawai*). This machine is registered as a primary node on the Active Network (ABone). We also have ten other machines (*mk00 – mk09*) that are registered on the ABone as well. We used this cluster of ABone nodes to generate background cross traffic while the video is playing. We run the player on a selected remote ABone node using the Anetd LOAD command from (*kawai*). We repeated the experiment on five ABone nodes, three in the US and two in Europe. All five nodes are shown in Table-1.

In all runs, the transcoder subscribes with iTCP for two events: REXMT (retransmit timer out event) and DUPACK (third duplicate acknowledgment event) see Table-2. Also, we always turn on the event notification property of the iTCP. The only controlled parameter that we changed was the reduction property of the signal handler. When the reduction flag was set (*symbiosis=on*), the signal handler invokes the event handler to reduce the bit rate of the decoder. Otherwise (when *symbiosis=off*), the signal handler just records the event type and time in a log file.

We repeated the experiment ten times with each remote ABone node from Table-1, five times in the (*symbiosis=on*) mode and five times in the (*symbiosis=off*) mode. In each run we recorded two log files, one on the transcoder side (*kawai*), and one on the player side (the remote ABone machine). We retrieved the latter log file using the Anetd GET command. The transcoder recorded the following information for each frame in the video stream: *frame number, departure time, target bits, actual bits, and SNR values for Y, U, and V blocks.* Also, when an event signal is received from the iTCP, the signal handler records its type and timing. On the player side, the log file only records the arrival time of each frame.

In the following discussion we will regard the (*symbiosis=on*) mode to resemble iTCP and the (*symbiosis=off*) to resemble classic TCP. We made this resemblance since the (*symbiosis=off*) mode adds only the event notification property to the TCP. This small overhead is irrelevant and can be ignored in the overall system performance analysis.

Subscribe flag (iTCP) = on
Event reception flag (EVENT) = on
Rate reduction flag (SYMBIOSIS) = on/off
Reduction Factor (ALPHA) = 0.55
Subscribed events = REXMT | DUPACK
Frame size = 176 x 120
Number of Frames = 1000

Table-2. Experiment and video parameters. Only the reduction flag (SYMBIOSIS) was changed in different runs.

4. Events and Video Quality Parameters

Table-3 shows several parameters to measure end-to-end performance at both the application and network levels on the five target ABone nodes. Part (a) of the table represents the results for the iTCP mode were symbiosis was applied and part (b) represents the results for the TCP classic mode. Each value in the table is an average of five runs on the specified ABone node. We show five parameters in Table-3: *average number of events, average SNR(Y) block per frame, average bits per frame, average time to transmit/play the entire video (1000 frames), and average frames per second.*

To facilitate comparison between the two cases, we converted each parameter from the Table-3 into a bar graph. We show these bar graphs in Figure-2. In each bar graph the x-axis represents the five target ABone nodes and the y-axis represents the measured parameter. First, we notice that the average number of congestion events for both TCP and iTCP modes on all ABone nodes were relatively close (1.4 on iTCP vs. 1.6 on TCP). This observation justifies the comparison and enables us to make the assumption that both modes were running under similar network conditions. Direct observation of these bar graphs reveals the advantage of the iTCP mode over TCP mode.

4.1. Video Quality and Frame Size

Next, we show the '*Average SNR*' per frame. Here we only show the Y block statistics. iTCP managed to reduce delay and jitter at some cost—video quality. Video quality was reduced temporarily while the network suffered from congestion and was restored to its normal rate when the network recovered. Therefore, the reduction in quality was momentarily and minimal. In the graph we see that quality is constant in the TCP-classic (or *symbiosis=off*) mode, but it was reduced in the iTCP mode. Most reduction happened in (*Italy*) and (*us-isi*) nodes since they were confronted with more congestion events than other nodes.

In the next graph, '*Average Bits per Frame*', we show how many bits on average there were in each frame. Like the previous SNR graph, TCP-classic carried the same number of bits per frame in all nodes since no reduction took place. To the contrary, iTCP reduced the bit rate and hence reduced the average frame size in the video.

4.2. Video Timing and Frame Rate

Next graph shows the '*Average Time per Video*'. Here we show the total time needed to transmit and play the video clip averaged for five runs per node. We think this parameter is important since it shows that iTCP managed to considerably reduce the overall delay of the video session even during severe congestion (e.g. with the '*Italy*' node there were about 50 seconds in favor of the iTCP mode). The last graph of Figure-2, '*Average Frames per Second*', shows the average frame rate of the video session on each node. This parameter is a direct consequence of the previous parameter (i.e. '*Average Time per Video*') and was calculated by dividing the number of frames in the video clip (1000 in our case) by the '*Average Time per Video*'.

5. Symbiotic Rate Control and Event Trace

Figure-3 shows the symbiotic frame rate transcoding for five runs on the target ABone nodes. The

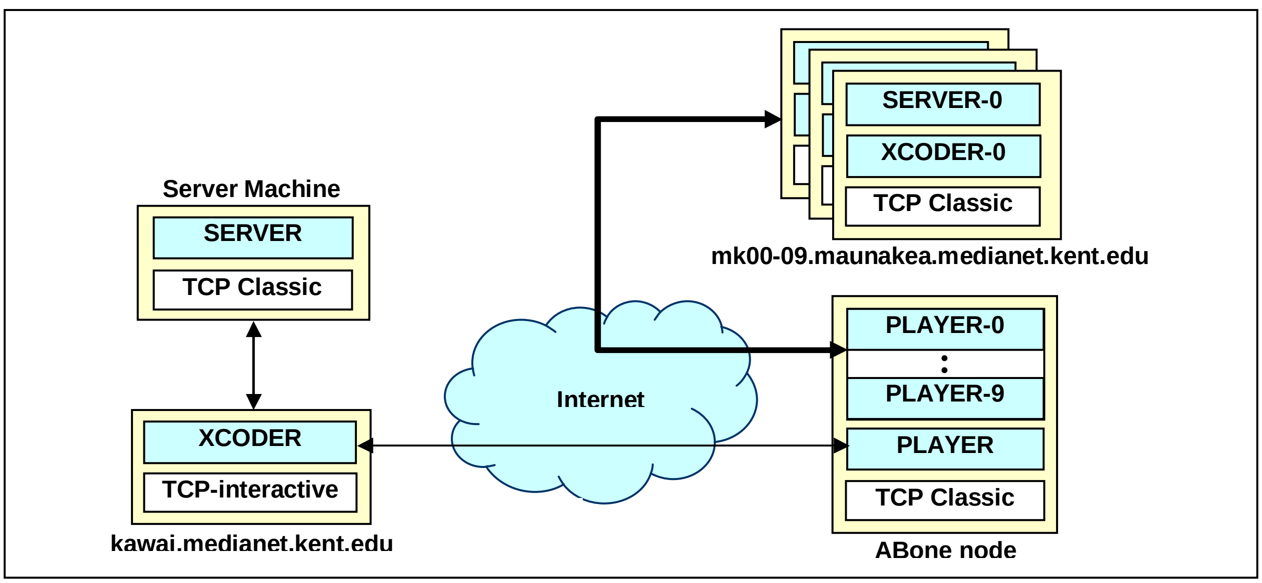

Figure-1. Experiment setup. The transcoder runs on the iTCP machine and the player runs on a remote ABone node. The mk00-09 cluster generates background cross traffic.

symbiosis occurred due to the joint rate specification at the rate control logic at the symbiosis unit and in the transcoder. Each case plots the incoming video frame sizes, the target *rate retraction ratio* specified by the symbiosis controller, and the resulting outgoing frame rate generated by the transcoder. Congestion events at the TCP (e.g. timeout event) resulted in the symbiosis unit to modify the rate according to the lazy-binary-back-off rule. A retraction ratio (Alpha) of 0.55 was used. Though, the final generation rate varied widely from frame to frame due to their frame type, but the general trend followed the specified target bit rate.

6. Observation at Application Level

In the above discussion we illustrated how the symbiosis mechanism worked from the video transport protocol (MPEG-2) and the network transport protocol (TCP) layers beneath it. In this plot we will illustrate how this mechanism appears from the very top-- in the application layer itself. An application receives and delivers uncompressed frames. The performance metric this end-system uses is the temporal and spatial quality difference between the transmitted and the reproduced uncompressed video frames at both ends. The underlying MPEG-2 transport protocol and the network layer TCP together provides the transport. The specific compression, windowing etc. and other detail mechanisms are external techniques to the end systems.

In Figure-4 each frame is plotted as a point in the video quality/frame delay plane. Each plot in the figure represents the average of five runs on the specified ABone node. As can be seen from the region of the two QoS distributions, in classical TCP (symbiosis=off), although frames have been generated with SNR quality ranging between 13-39 dB, but many of these frames were lost in transport, and were never delivered. In contrast, the proposed iTCP (symbiosis=on) can deliver all the frames guaranteed at 10-38 dB quality. All plots show the Y block quality. Table-4 shows the average for all Y, U, and V blocks. Fundamentally, what TCP interactive has offered is a qualitatively (as opposed to the quantitative improvements offered by any unaware solution) new empowering mechanism, where the catastrophic frame delay can be traded off for acceptable reduction in SNR quality.

Conclusions and Current Work

In this report, we have presented a case of rate symbiosis mechanism in line with current advances in TCP friendly systems. We have presented the case through a simple ‘interactive’ generalization of the classical transport control protocol, and a novel implementation of a symbiotic MPEG-2 transcoder. We collected the results of our experiment by running the video session on the global Active Network (ABone) testbed.

In previous discussion we have demonstrated the case of quality conformant congestion control for time-sensitive video traffic. The approach exposed the overall advantage of network ‘*friendly*’ applications. However, it also departs significantly from the mainstream TCP friendly systems that have been suggested recently in two senses; First, it does not add any new major component in network software structure. One of the principal strength of the proposed scheme is its relative simplicity at network layers –yet its effectiveness. It only expects some form of interactivity directly from the concerned network protocols as a general interface feature. Thus, there is no expectation of (or conflict with) additional services (such as combined congestion control from multiple applications).

Secondly, the applications do not have to be designed dependent on other auxiliary indirect probing tools or network utilities, nor it excludes their use when available. Some of the information measured by the auxiliary tools suggested by other approaches might be already available (or are being estimated/tracked) at lower layers anyway. At least this is the case with TCP congestion. The direct protocol interactivity we propose thus seems to be the logical path that can avoid potential duplication of efforts.

ABone node	Num. Of Events	SNR per frame	Bits per frame	Time per video	Frames per Second
fokus (denmark)	2	21.942224	49737.387	115.615629	8.70646714
galileo (italy)	1.2	18.423304	36672.216	107.585495	9.34654553
columbia (usa)	1.6	22.51238	50173.551	101.357553	9.90596598
princeton (usa)	1.2	22.463844	50331.645	110.164151	9.22516667
isi (usa)	1	20.712062	47694.519	107.421659	9.37435276
AVERAGE	**1.4**	**21.210763**	**46921.864**	**108.42889**	**9.3116996**

(A)

ABone node	Num. Of Events	SNR per frame	Bits per frame	Time per video	Frames per Second
fokus (denmark)	1	23.5559	54672.969	141.7599768	7.903423414
galileo (italy)	2	23.5559	54672.969	157.2548856	6.554845167
columbia (usa)	1.6	23.5559	54672.969	112.2868738	8.964446532
princeton (usa)	1.4	23.5559	54672.969	112.419281	8.978581526
isi (usa)	2.2	23.5559	54672.969	122.7066482	8.238424173
AVERAGE	**1.64**	**23.5559**	**54672.969**	**129.28553**	**8.1279441**

(B)

Table-3. Network and video parameters. Table (A) shows the results of the iTCP runs, while table (B) shows the results of the classic TCP-classic runs.

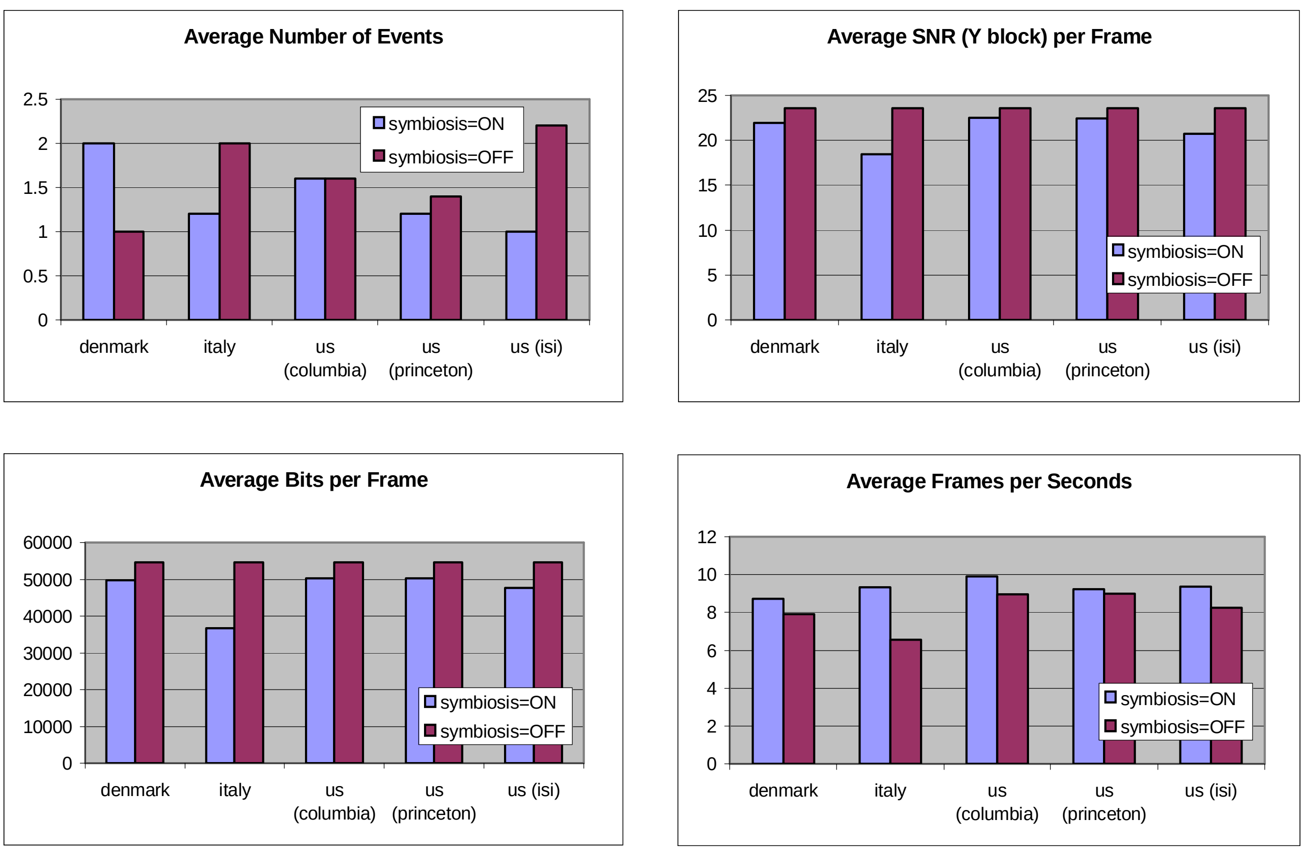

Figure-2. Video quality parameters from Table-3. Each parameter is shown as a separate bar graph to facilitate comparison between the two modes of experiment (sym=ON|OFF).

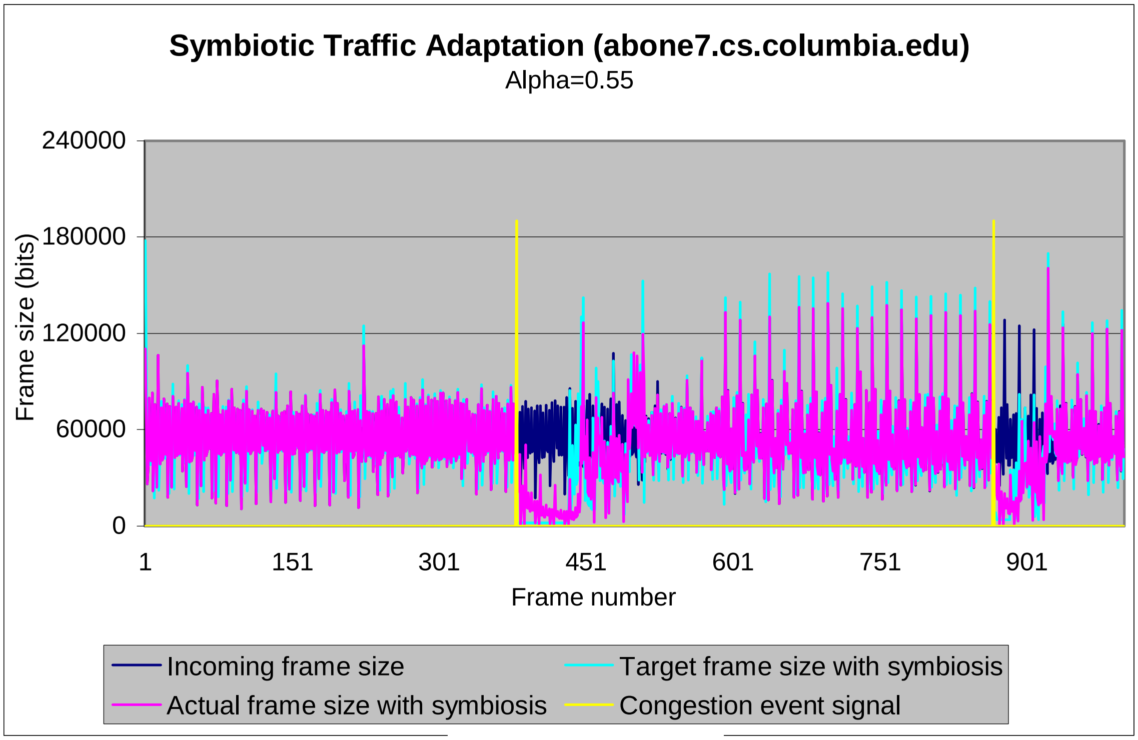

Figure-3 (A)

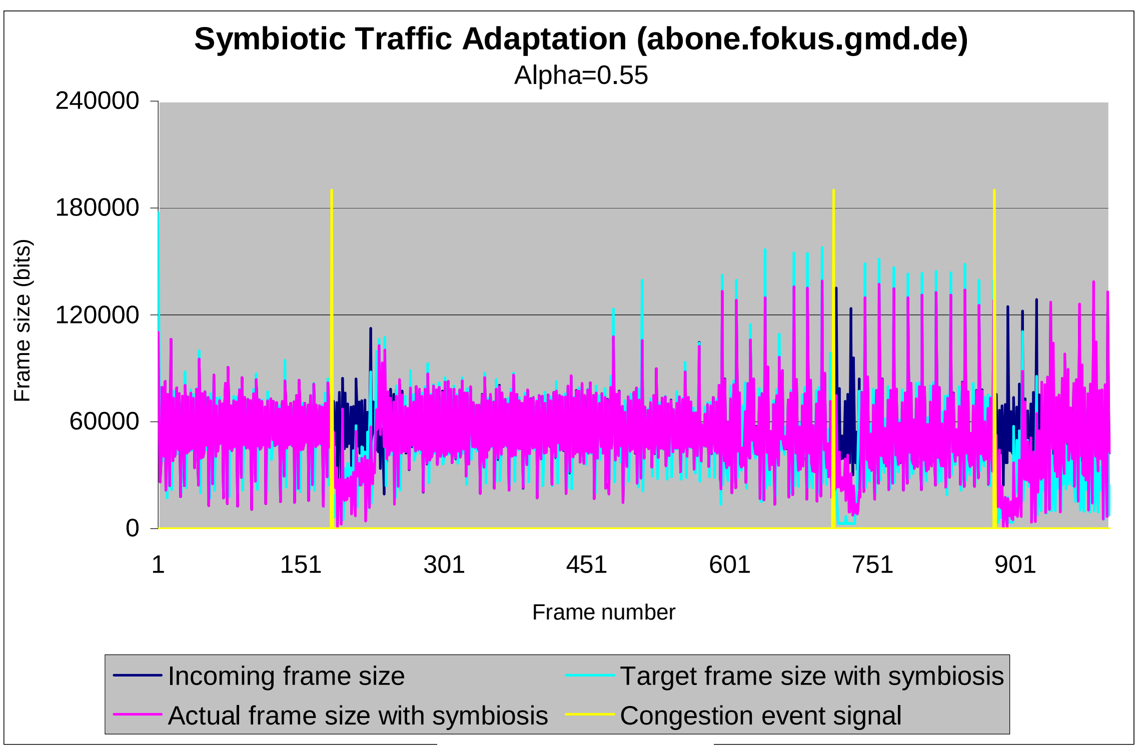

Figure-3 (B)

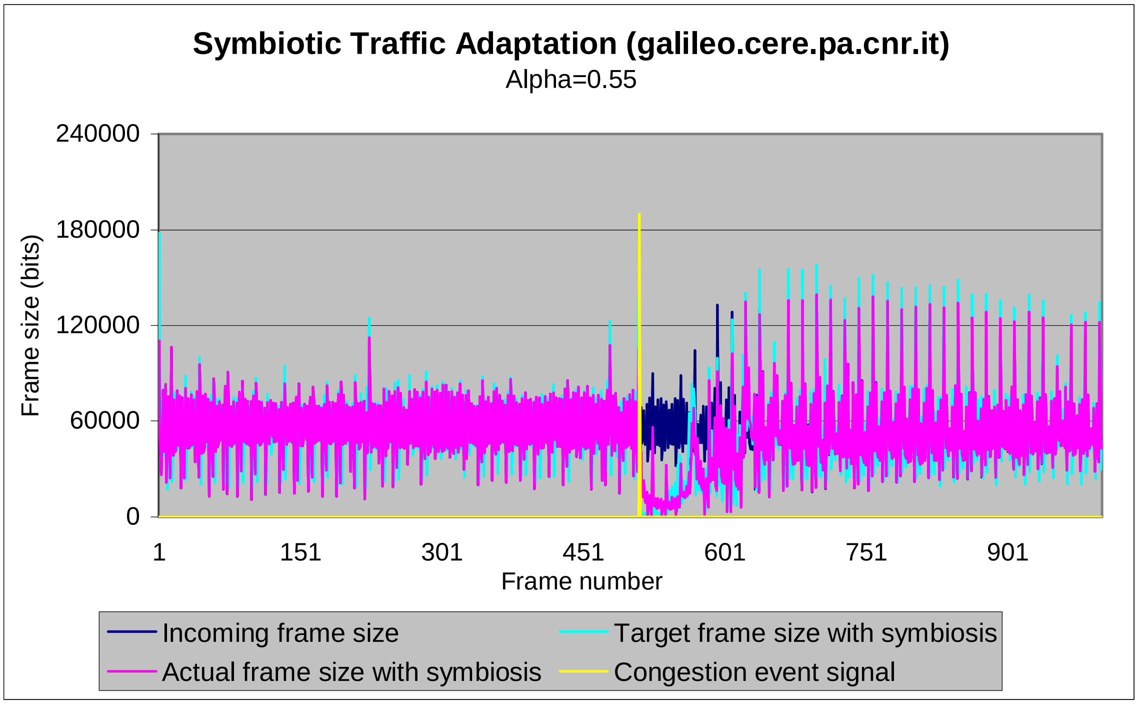

Figure-3 (C)

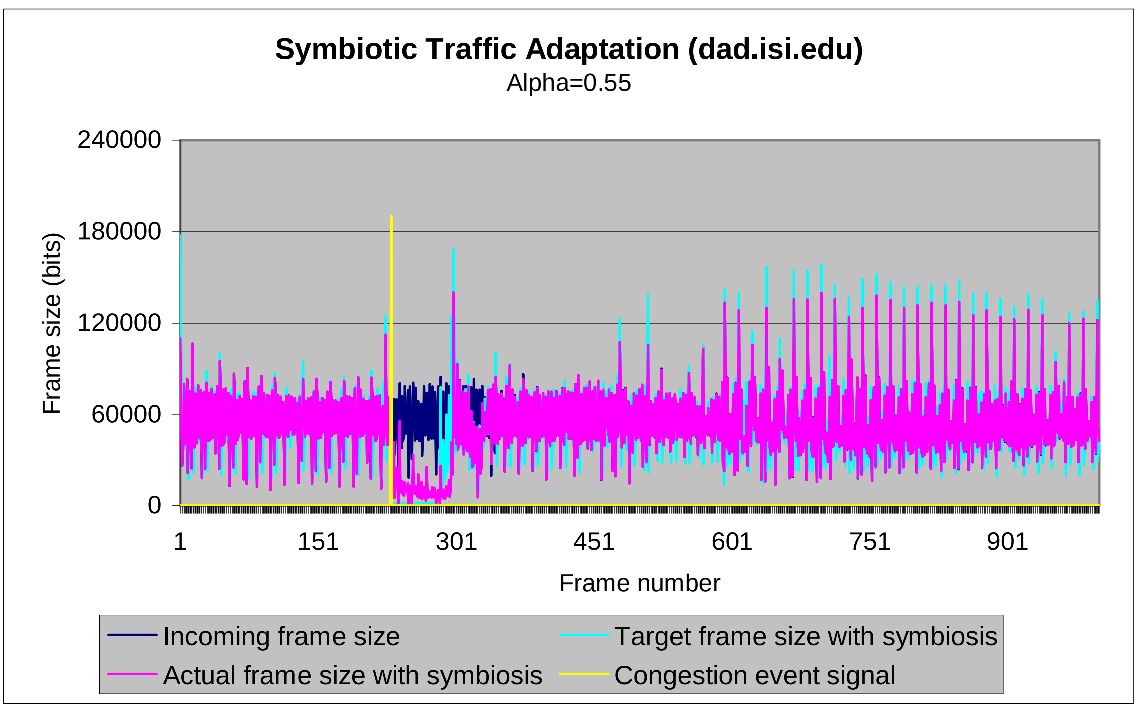

Figure-3 (D)

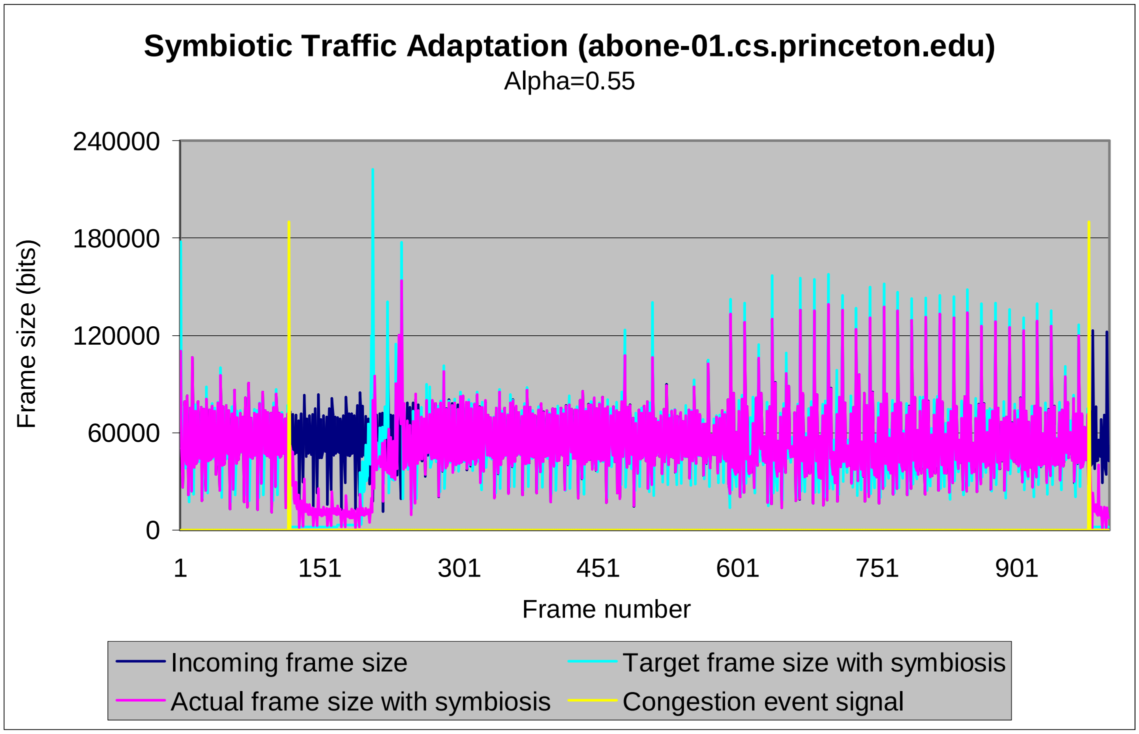

Figure-3 (E)

Figure-3. The binary back-off rate reduction in transcoder. Plots the incoming frame size, the event driven target rate (retraction ration) specified by the symbiosis unit, and the resulting output rate from the transcoder.

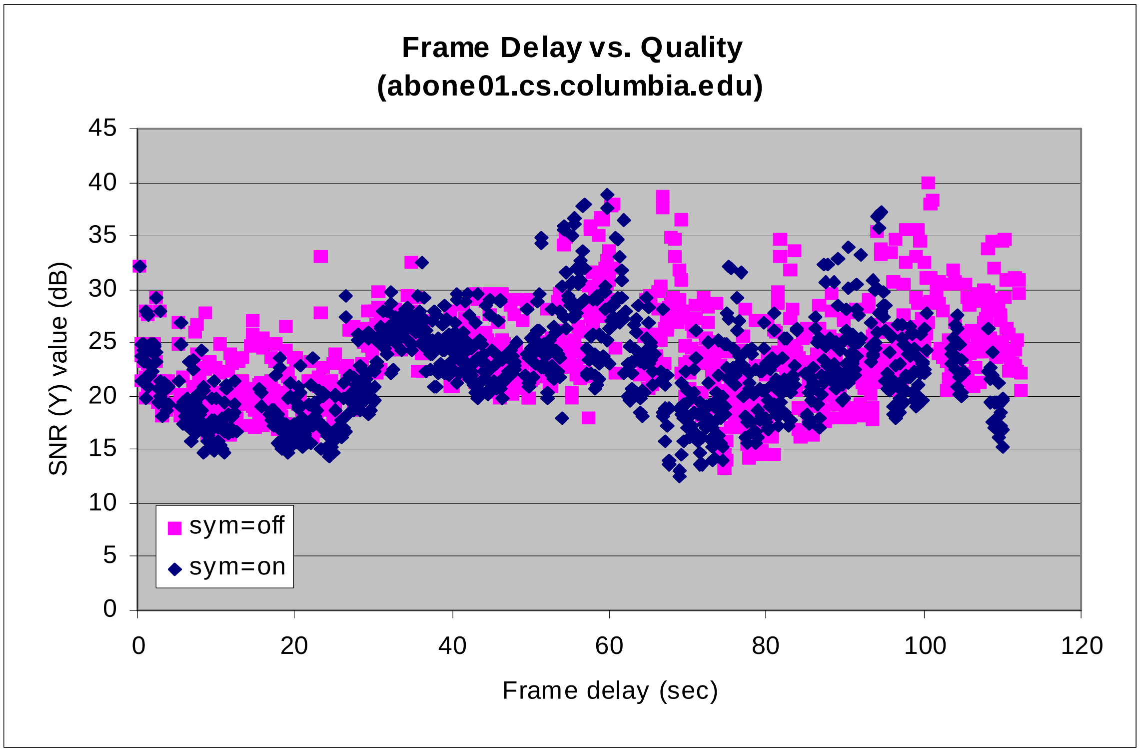

Figure-4 (A)

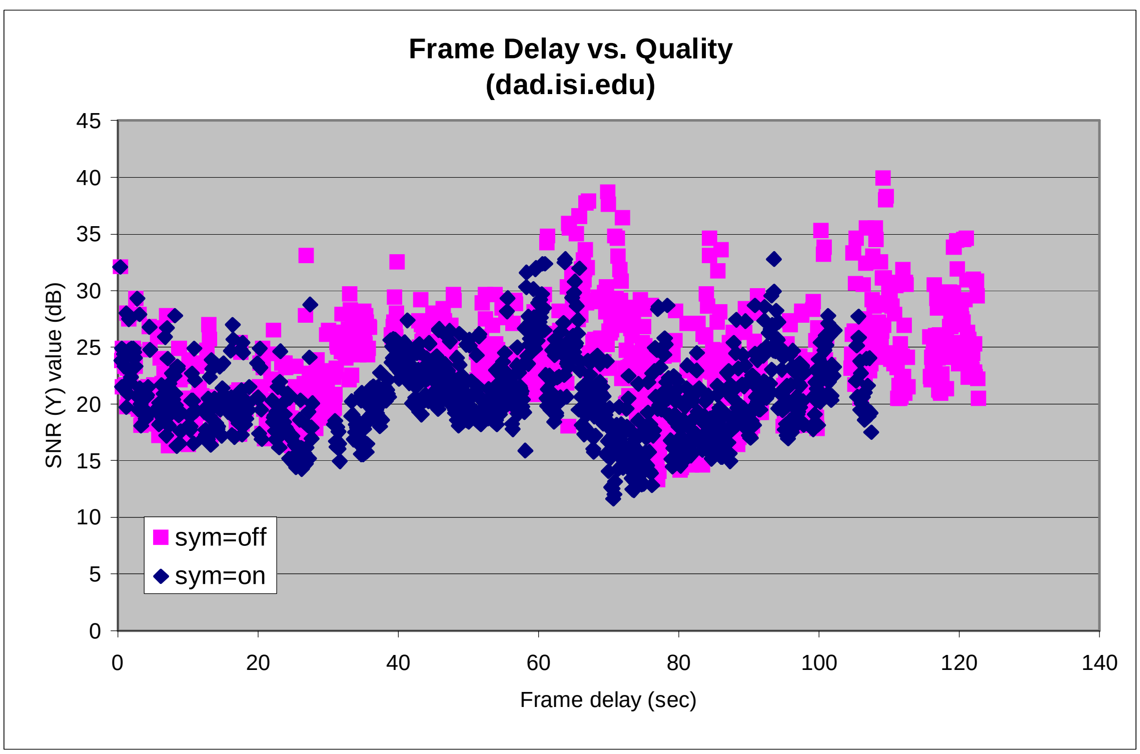

Figure-4 (B)

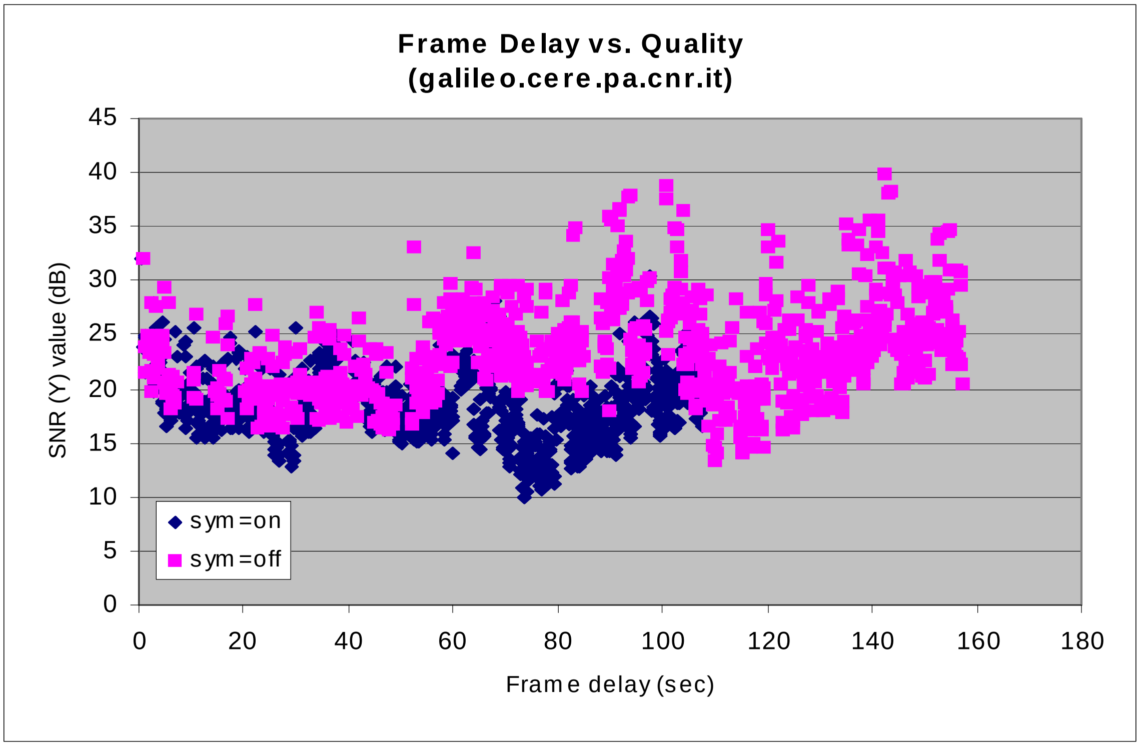

Figure-4 (C)

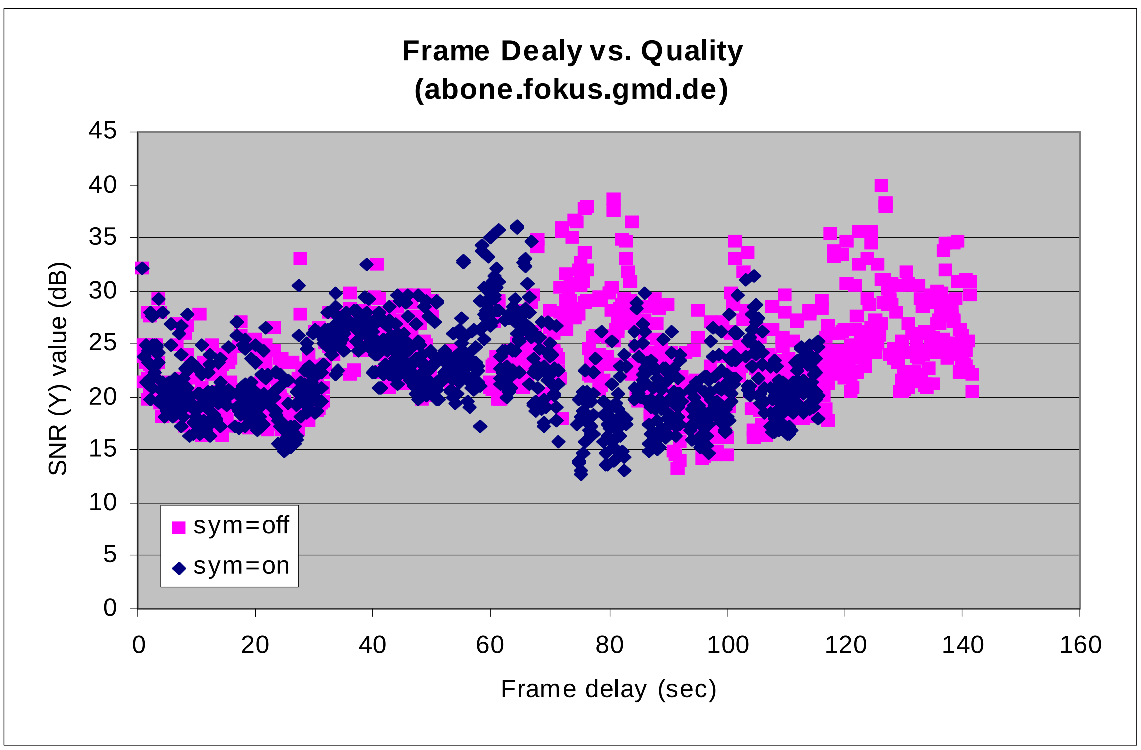

Figure-4 (D)

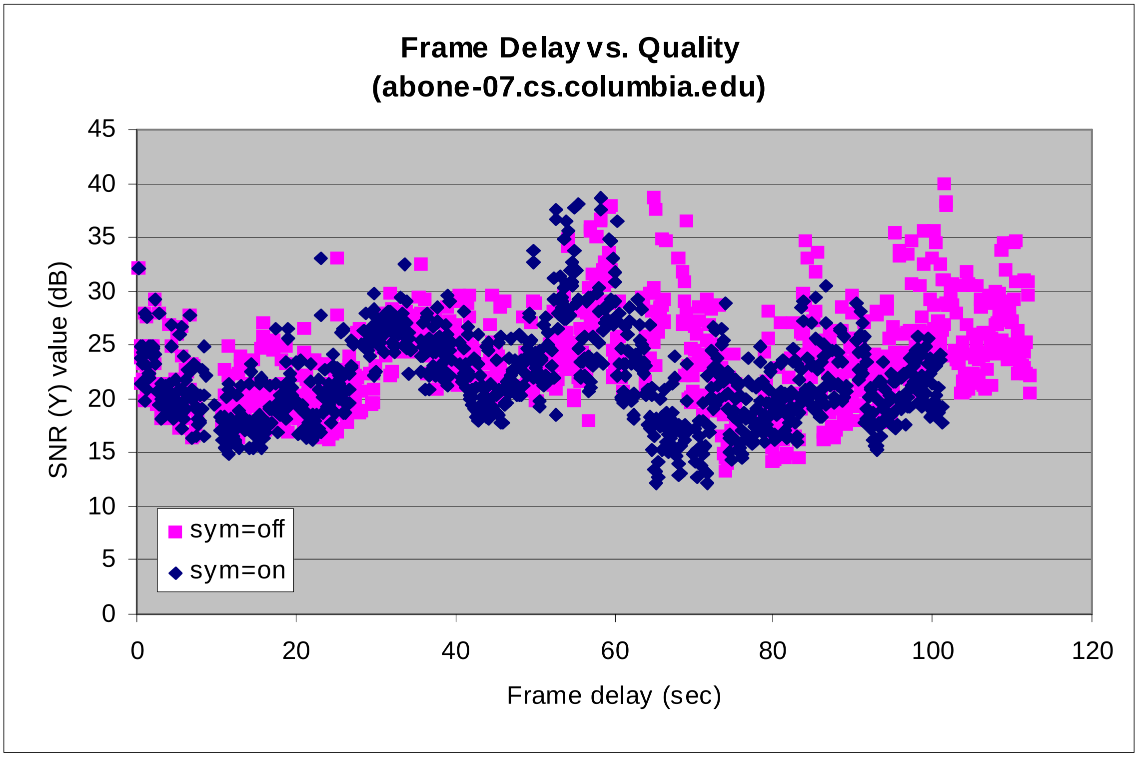

Figure-4 (E)

Figure-4. Frame delay versus video quality (Y block). Shows the quality, delay tradeoff by the iTCP. The iTCP dramatically reduced frame delivery delay by controlled tradeoff of the SNR quality.

abone7.cs.columbia.edu			
	Y	**U**	**V**
	23.5559	11.42286	11.89164
	22.27272	10.73152	11.17327
	20.91348	9.93609	10.29408
	22.2639	10.79802	11.37637
	19.9173	9.434132	9.625847
Average	**21.78466**	**10.46453**	**10.87224**

abone.fokus.gmd.de			
	Y	**U**	**V**
	21.7502	10.4313	10.88387
	22.55195	11.00852	11.37276
	22.48072	10.8563	11.33524
	20.32865	9.53193	9.7699
	22.5996	10.98443	11.52106
Average	**21.94222**	**10.5625**	**10.97657**

galileo.cere.pa.cnr.it			
	Y	U	**V**
	14.20791	6.309307	6.57389
	18.78038	8.54067	8.73877
	17.55128	7.75856	7.91798
	18.88435	8.58182	8.78481
	22.6926	10.91146	11.37236
Average	**18.4233**	**8.420363**	**8.677562**

dad.isi.edu			
	Y	**U**	**V**
	22.6875	**10.9431**	**11.38217**
	20.24505	9.4674	9.70093
	14.81236	6.360895	6.513461
	22.2595	10.78451	11.22879
	23.5559	11.42286	11.89164
Average	**20.71206**	**9.795753**	**10.1434**

Table-4. The average picture quality for Y, U, and V components on five runs of iTCP.

References

[ABCS00] D. Andersen, D. Bansal, D. Curtis, S. Seshan, and H. Balakrishnan, System Support for Bandwidth Management and Content Adaptation in Internet Applications, Proceedings of the Fourth Symposium on Operating Systems Design and Implementation, OSDI 2000, October 23-25, 2000 San Diego, California.

[AlPa99] Allman, Paxson, et al. TCP Congestion Control, RFC 2581, April 1999.

[BaRS99] Balakrishnan, H., Rahul, H., and Seshan, S., An Integrated Congestion Management Architecture for Internet Hosts," Proc. ACM SIGCOMM, Cambridge, MA, September 1999.pp.175-187.

[BrGM99] Hector M. Briceño, Steven Gortler and Leonard McMillan, NAIVE--network aware Internet video encoding, Proceedings of the seventh ACM international conference on Multimedia, October 30 - November 5, 1999, Orlando, FL USA, Pp 251-260.

[BrOP94] Brakmo, L.S., O'Malley, S.W., and Peterson, L.L, TCP Vegas: New Technique for Cogestion Detection and Avoidance, Proc. SIGCOMM' 94 Conf. ACM, pp-24-35, 1994.

[GuTe98] Wetherall, Guttag, Tennenhouse, "ANTS: A Tool kit for Building and Dynamically Deploying Network Protocols", IEEE OPENARCH'98, San Francisco, April 1998. Available at: http://www.tns.lcs.mit.edu/publications/openarch98.html

[ISO96] Information Technology- Generic Coding of Moving Pictures and Associated Audio Information: Video, ISO/IEC International Standard 13818-2, June 1996.

[Jac88] Jacobson, V, "Congestion Avoidance and Control", Proc. SIGCOMM' 88, Conf, ACM, pp-214-329, 1988.

[Jac90] Jacobson, V., "Modified TCP Congestion Avoidance Algorithm," end2end-interest mailing list, April 30, 1990.

[KeWi00] Jun Ke and Carey Williamson / University of Saskatchewan, Towards a Rate-Based TCP Protocol for the Web, Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000.

[KhGu01] Javed I. Khan, Q. Gu, Network Aware Symbiotic Video Transcoding for Instream Rate Adaptation on Interactive Transport Control, IEEE International Symposium on Network Computing and Applications, IEEE NCA' 2001, October 8-10, 2001, Cambridge, MA, pp.201-213

[KHFH96] Keesman, Gertjan, Hellinghuizen, Robert Hoeksema, Fokke Heideman, Geert, Transcoding of MPEG bitstreams Signal Processing: Image Communication, Volume: 8, Issue: 6, pp. 481-500, September 1996,

[KYGP01] Javed I. Khan, Seung Su Yang, Qiong Gu, Darsan Patel, Patrick Mail, Oleg Komogortsev, Wansik Oh, and Zhong Guo Resource Adaptive Netcentric Systems: A case Study with SONET- a Self-Organizing Network Embedded Transcoder, Proceedings of the ACM Multimedia 2001, October 2001, Ottawa, Canada, pp617-620

[KhY01] Javed I. Khan & S. S. Yang, Resource Adaptive Nomadic Transcoding on Active Network, International Conference of Applied Informatics, AI 2001, February 19-22, 2001, Insbruck, Austria, [available at URL http://medianet.kent.edu/, also mirrored at http:// bristi.facnet.mcs.kent.edu/medianet] (accepted).

[KhGR02] Javed I. Khan, Qiong Gu and Raid Zaghal, Symbiotic Video Streaming by Transport Feedback based quality rate selection, Proceedings of the 12th IEEE International Packet Video Workshop 2002, Pittsburg, PA, April 2002, http://www.pv2002.org .

[KhZ03R1] Javed Khan and Raid Zaghal, "Event Model and Application Programming Interface of TCP-Interactive", Technical Report TR2003-02-02, Kent State University, [available at URL http://medianet.kent. edu/ technicalreports.html] March 2003.

[KhZ03R2] Javed Khan and Raid Zaghal, "Jitter and Delay Performance Analysis of TCP Interactive based World Wide Video Streaming over ABone", Technical Report TR2003-03-03, Kent State University, [available at URL http://medianet.kent. edu/ technicalreports.html] March 2003.

[KPOY01] Javed I. Khan, Darsan Patel, Wansik Oh, Seung-su Yang, Oleg Komogortsev, and Qiong Gu, Architectural Overview of Motion Vector Reuse Mechanism in MPEG-2 Transcoding, Technical Report TR2001-01-01, Kent State University, [available at URL http://medianet.kent. edu/ technicalreports.html, also mirrored at http:// bristi.facnet.mcs.kent.edu/medianet] January, 2001.

[Mpeg00] MPEG-2 Test Model 5, [URL: http://www.mpeg.org/MPEG/MSSG/tm5/, last retrieved December 25, 2000]

[PeDa00] 19 L. L. Peterson and B. S. Davie. Computer Networks, 2nd edition, Morgan-Kaufmann, 2000.

[PrCN00] Prashant Pradhan, Tzi-cker Chiueh and Anindya Neogi Aggregate TCP Congestion Control Using Multiple Network Probing, Proceedings of the The 20th International Conference on Distributed Computing Systems, ICDCS 2000. 2000.

[ReHE00] Reza Rejaie,Mark Handley, Deborah Estrin, Architectural Considerations for Playback of Quality Adaptive Video over the Internet, Proceedings of the IEEE International Conference on Networks (ICON'00), 2000.

[SiWo98] Dorgham Sisalem, Adam Wolisz, Towards TCP-Friendly Adaptive Multimedia Applications Based on RTP, Proceedings of the The Fourth IEEE Symposium on Computers and Communications, 1998.

[Tene96] Tanenbaum, A. Computer Networks, 3rd edition, Prentice Hall, 1996.

[TSSW97] Tennenhouse, D. L., J. Smith, D. Sincoskie, D. Wetherall & G. Minden., "A Survey of Active Network Research", IEEE Communications Magazine, Vol. 35, No. 1, Jan 97, pp 80-86

[Wolf97] Bernd E. Wolfinger, On the potential of FEC algorithms in building fault-tolerant distributed applications to support high QoS video communications, Proceedings of the sixteenth annual ACM symposium on Principles of distributed computing , 1997, Pages 129 – 138