Performance Comparison Of Video Transport Over ATM and ServerNet Interconnects

Ashfaq Hossain                 Sung-Mo Kang                  Bob Horst
University of Illinois,        University of Illinois,        Tandem Computers Inc.,
Urbana, IL 61801.              Urbana, IL 61801.               Cupertino, CA 95014.

Abstract

We discuss the implementation of iPOINT Video Server and Video Client applications along with rate-control feedbacks and network loss handling mechanisms. We report experimental results for ATM and ServerNet, using 60MHz SparcStation20s (SS20), 30MHz SparcStation10s (SS10) and 100 MHz Pentium (P5) systems as video servers. Our study compares CPU utilization and corresponding video throughput of these platforms, running Solaris 2.5 and WindowsNT 3.51, respectively. We show how these network interfaces affect the behavior of these hosts as video servers and thereby the quality of served video.

1 Introduction

We have investigated distribution of pre-compressed digital video over ATM and ServerNet (from Tandem Computers Inc.). We present experimental results for video throughput and the corresponding CPU utilization for different end-host platforms which implement these networks. We identify the strengths and weaknesses of such networks by observing the end-host behaviors. All the experimental runs for ATM reported in this work are based on fully functional Video Server (VS) and Video Client (VC) applications developed in the iPOINT (Illinois Pulsar-based Optical Interconnect) testbed with an 800Mbps ATM switch. The results for ServerNet are from corresponding runs of these applications on Tandem Computer's ServerNet interconnect.

2 Interworking of iPOINT Video Server and Client Applications

The iPOINT VS and VC implementations draw on UDP/IP socket interface and on Fore System's ATM APIs (Application Programmer's Interface). On receiving a request, the server creates a Compressed Image Sequence (CIS) for a specific type of stream (MJPEG or MPEG-1), mmaps the video file to the user memory, then gradually reads an integral number of frames into the CIS. These frames are then transferred as IP packets or AAL5 frames. Any rate-adaptation (e.g. SLOWDOWN, SPEEDUP) or repair-request (RETRANSMIT) feedback messages from the VCs are handled by the VS with high priority. The packet sizes are limited to 9180 bytes for AAL5 frames (maximum MTU) or to the size set by the setssockopt() call for IP transmissions (maximum 64KBytes). Any network loss is handled by retransmission requests sent from the VCs to the VS.

The VC implements a ring-buffer on its end to insert the received compressed frames - which can be sent by the VS as an entire video frame in a single packet or can be broken up into smaller parts (at the application layer). The ring-buffer consists of n nodes. Each node can hold up to 2 compressed MJPEG frames (for a frame size of 320 x 240 pixels) and a single I-P-B frame combination for MPEG-1 stream. For current implementation, the total memory for the ring buffer is just about 1MByte. The independent Receive() and Decompress() threads chase each other in the ring and can send appropriate rate-control feedbacks to the VS depending on the distances (in ring nodes) between them. The VC is responsible for detecting lost portions of a transmitted video frame. When such a situation is detected, the client dynamically creates a RepairRequest() thread which is responsible for sending a bit array to the server (RETRANSMIT message). This bit array indicates which portions of the entire video frame has not been received (if frames are being transmitted in parts). Out-of-order arrivals (possible in UDP/IP protocol and also due to repair packets sent by the VS) are placed in appropriate positions in the ring-buffer - any duplicate arrivals are quietly discarded.

3 Performance Over ATM Testbed

We first consider the experiments in which there is no rate control feedback from the VC to the VS. The VC maintains a decompression/display rate which matches VS's transmission rate of 17fps (for MJPEG) and 10fps (for MPEG-1) up to certain points in time. After these, VC's performance degrades sharply (to 1fps for MJPEG and to 4fps for MPEG-1) and remains degraded. This occurs due to increased OS activity for buffering incoming packets into the finite kernel memory, as the application space buffer becomes full. This takes away scarce CPU resource from software-based decompression resulting in degraded playback frame-rate. Without feedback this degradation never improves. In order to avoid the situation, the VC's send feedback messages to the VS to control the transmission rate dynamically. We implement
two approaches to feedback: feedback after ring buffer becomes full and performance degrades (kernel feedback) and feedback before the ring buffer becomes full (when the buffer fullness reaches certain high water mark). In the first approach, VC's playback rate improves after degradation (resulting in jitter). In the second approach, playback is smooth as degradation is avoided by not allowing the ring-buffer to become full.

4 ServerNet Video Transport

ServerNet is a new System Area Network (SAN) that has been designed specifically as a reliable, high-speed connection among processors and I/O devices in a cluster. It is a wormhole-routed multistage network formed by point-to-point connections to six-port router ASICs. Reliability is assured by self-checking routers and network adapters, support for dual fabrics, and link-level flow control to eliminate packet loss on heavily loaded networks. In order to transfer $N$ bytes of data from machine A to machine B, the Interface Layer of the ServerNet protocol writes a chain of BTE (Block Transfer Engine) descriptors in the main memory (MM) of machine A. Each BTE descriptor is responsible for $4K$ Bytes (max) of data (for $N$ bytes of total transfer, $\lceil \frac{N}{4K} \rceil$ BTE descriptors are created). Each descriptor contains the address of these $4K$ Byte data, destination node ID, and the address of the next BTE descriptor (thus establishing the chain). After ServerNet driver initializes the transfer, the host CPU is relieved of any activity related to the transfer of $N$ bytes over ServerNet. This results in considerable lowering of the CPU utilization for large packet transfers, relieving the CPU for other related or non-related activities. It is very important to realize (especially for video data transfer) that ServerNet packets are never lost due to network congestion. If a congestion is detected, a BUSY command (similar to the SLOW-DOWN feedback from the video client) is sent by the ServerNet hardware to the source Network Interface Card (NIC) through the back channel, thereby reducing Server's transmission rate. Currently, ServerNet implements a 9-bit channel in each direction with a driver clock of 50MHz.

5 ServerNet vs ATM Performance

We present a comparison of ServerNet and ATM performance for video service in Table 1. The columns indicate maximum data throughputs, packet sizes and CPU utilization at throughput saturation, number of video streams served. System specifications are indicated in column 1. We varied packet sizes from 10Bytes to 64KBytes for throughput and CPU utilization measurements. Maximum throughputs are found to be higher in ServerNet (20.3 MBps vs 10.8 MBps). It is important to realize that at the throughput saturation points, the SS20 and SS10 with ATM are entirely CPU bound—there they have no CPU resource available to perform any other functionality. On the other hand, the P5 implementing ServerNet has about 91% of its CPU resource available—this can be used for additional data transfers across other NICs fitted in other slots provided the PCI bus saturation does not occur. Even though ServerNet measurement is performed on higher clock rate 100MHz P5, some benchmark tests put SS20/60 machine at a better performance due to its dual issue (superscalar) instructions per cycle (CPI < 1). As regards SPEC numbers for these two processors, the P5 performance is between 0.9x and 2.0x that of SS20 (data obtained from [http://infopad.EECS.Berkeley.edu/CIC/](http://infopad.EECS.Berkeley.edu/CIC/)).

ServerNet has much lower CPU overhead than ATM because the hardware directly implements guaranteed, in-order delivery of packets even when the network is heavily loaded. There is no need for software to detect loss of packets, or to compute checksums. Once the BTE chain is initiated, the transfer is controlled completely by the hardware, and no more software execution cycles are consumed by either the client or the server until the transfer is completed. In effect, this means that ServerNet hardware implements levels 1-4 of the OSI stack, while the ATM hardware provides only an unreliable datagram service and requires levels 3 and 4 to be performed by the software.

We also compare the maximum number of MJPEG threads which can be served over ServerNet and ATM interconnects. The video stream we used was $320 \times 240$ pels, 9KB/frame, transmitted at 18-20fps (1.5Mbps per stream). Total throughput from all the video threads combined reasonably matches the maximum deliverable throughput from these NICs at the packet sizes used (except for thread overheads and global variable manipulation through exclusion locks). The number of MJPEG threads that can be served for the hosts are 70 for ServerNet and 47 and 20 over ATM (SS20, SS10), respectively. It should be noted from Table 1 that a higher number of video streams could be served over ServerNet if a video packet size larger than 20KB was used. For IP-over-ATM, sending a video packet larger than 10KB would not help as saturation occurred for this size of packets.

An extended version of this paper can be anonymously ftp-ed from ipoint.vlsi.uiuc.edu in /pub/IEEEMM97/ieee.mm97.ps.

<table>
<thead>
<tr>
<th>Network</th>
<th>Maxm Tput, MBps</th>
<th>Saturation Pkt Size, KB</th>
<th>CPU Util.</th>
<th>MJPEG Streams Served</th>
<th>Video Packet Size(app.), KB</th>
<th>Video Tput, MBps</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM (SS20/60MHz)</td>
<td>10.8</td>
<td>10</td>
<td>100%</td>
<td>47</td>
<td>10</td>
<td>9.0</td>
</tr>
<tr>
<td>ATM (SS10/30MHz)</td>
<td>5.1</td>
<td>10</td>
<td>100%</td>
<td>22</td>
<td>10</td>
<td>4.3</td>
</tr>
<tr>
<td>ServerNet (P5/100MHz)</td>
<td>20.3</td>
<td>64</td>
<td>9%</td>
<td>70</td>
<td>20</td>
<td>13.5</td>
</tr>
</tbody>
</table>

Table 1: ServerNet and ATM Performance Comparison For Video Transport.