Experiences from Monitoring a Hybrid Fiber-Coaxial Broadband Access Network

Ed Perry, Srinivas Ramanathan
Internet Systems & Applications Laboratory
HPL-98-67
April, 1998

E-mail: [edp,srinivas]@hpl.hp.com

broadband data networks, access networks, network monitoring & management, hybrid fiber coaxial network technology

Hybrid fiber-coaxial (HFC) technology is emerging as one of the means of delivering high-speed data services to the home. Cable operators worldwide are deploying HFC networks to enable subscriber accesses to the World Wide Web, to Email, Chat rooms, and other data services at speeds ranging up to 10 Mbps. As cable networks, which have hitherto been deployed mainly for analog television broadcasts, begin to support digital on-demand delivery of data services, a number of challenges remain to be tackled. One of the key challenges is to effectively monitor the cable network in order to proactively detect and fix network problems before subscribers notice and complain about them. In this paper, we describe experiences from monitoring a cable network for several months. Using real-world examples, we illustrate that even with minimal monitoring capabilities built-into the current generation of cable modems and other cable networking equipment, it is possible to effectively monitor the status and performance of the cable network and detect problems proactively. Since they rely on built-in instrumentation, our monitoring tools provide an extremely low-cost, attractive alternative to the more expensive, external instrumentation that is currently deployed in many cable networks.

Internal Accession Date Only

1 Introduction

Motivated by the phenomenal increase in popularity and growth of the Internet, cable operators worldwide are beginning to deploy Hybrid Fiber Coaxial (HFC) technology to offer high-speed access to residential subscribers. Using efficient data modulation schemes such as Quadrature Amplitude Modulation (QAM) [5], these HFC access networks are capable of transporting tens of Megabits of information per second, thereby offering nearly thousand-fold increase in access bandwidth to residential subscribers compared to conventional telephone dial-up networks [4]. Using cable modems in their homes, residential subscribers connect to the HFC network and access a variety of Internet applications including the World Wide Web, Email, Chat rooms, and News.

As HFC networks, which have hither to been deployed mainly for analog television broadcasts, begin to support digital on-demand delivery of data services, a number of challenges remain to be tackled. One of the key challenges is to effectively monitor the HFC network in order to proactively detect and fix network problems before subscribers notice and complain about them. Until now, external special-purpose instruments such as spectrum analyzers installed at the cable network headend, end-of-line monitors that are permanently deployed out at the cable network periphery, and portable signal-level monitors have been the predominant means of monitoring and troubleshooting HFC network problems. In this paper, we describe experiences from monitoring data services deployed over an HFC network for several months. We describe a simple, yet effective application called *hfcmon* that we have developed for monitoring HFC networks. *hfcmon* utilizes monitoring capabilities built-into the HFC data network equipment (e.g., cable modems, headend cable termination units, subscriber PCs, etc.) themselves to proactively detect and alert operations personnel about network problems that may be impacting service to subscribers. Using real-world examples[1], we illustrate that even with minimal monitoring capabilities built-into the current generation of cable modems and other HFC data networking equipment, it is possible to effectively monitor the status and performance of the HFC network and detect problems.

Since it relies on in-built instrumentation, *hfcmon* provides an extremely low-cost, attractive alternative to the more expensive, external instrumentation that is currently deployed in many HFC networks. Since many other emerging broadband access technologies such as Asymmetric Digital Subscriber Line (ADSL) and Wireless Local Multipoint Distribution Service (LMDS) share many of the characteristics of HFC networks, the basic principles of *hfcmon* are applicable to these other local loop technologies as well.

The rest of this paper is organized as follows: In Section 2, we describe the architecture of the HFC network under consideration. In Section 3, we discuss the specific management needs of HFC networks. Sections 4 and 5 describe quality of service (QoS) and usage monitors that are incorporated in *hfcmon*. Finally, Section 6 summarizes the contributions of this paper.

1. The examples presented in this paper were specifically chosen to reflect the effectiveness of hfcmon in various scenarios. These examples are not to be construed to be an indication of the typical performance of the HFC data network being monitored.

2 Hybrid Fiber Coaxial Network Architecture

To set the context for the rest of the paper, we depict the architecture of the HFC network that served as a real-world testbed for our study in Figure 1. This network, which was the first of its kind to support broadband data services on a large scale, passed several hundred-thousand homes in a large US metropolis. The main components of this architecture are:

- A ***Server Complex*** located in the cable network headend houses content servers that are repositories of information content (e.g., news, advertisements, yellow pages, etc.) relevant to the local subscriber community. In addition, a local cache of frequently accessed information from the Internet is maintained at caching proxy servers. The server complex also includes servers for applications that subscribers use (e.g., E-mail servers, News servers, etc.), subscriber management servers that handle subscriber authentication and interfaces to billing systems, routers that forward packets to and from the Internet and other external networks, and firewalls that control access to and from the server complex. A monitoring station in the server complex is the host that executes the monitoring software described in this paper.

  In view of the large switching speeds necessary to support over a hundred thousand subscribers, FDDI switching and distribution technology is used. The Internet Protocol (IP) is the network layer protocol of choice for this network, in order to seamlessly integrate the access network with the Internet, thereby making the entire suite of Internet applications available to subscribers.

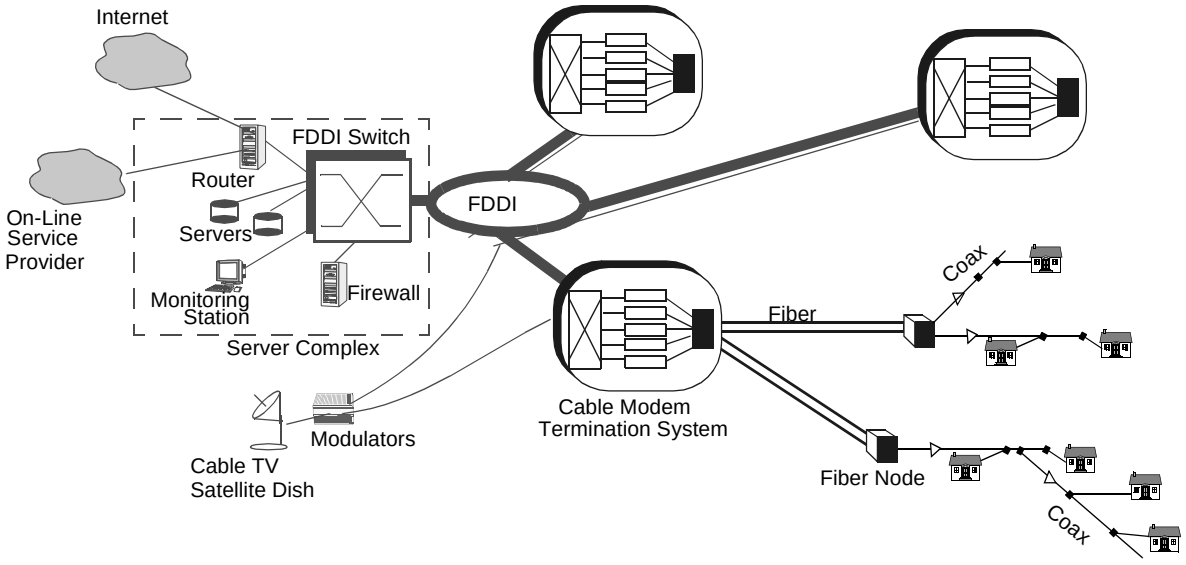


Figure 1. Configuration for providing broadband data services over an HFC network

- ***Cable Modem Termination Systems*** *(CMTSs):* Transmission of information to and from the server complex via the fiber-coaxial network is enabled by Cable Modem Termination Systems. Frequency division multiplexing is used for communication over the HFC network. While analog cable television channels occupy the range of spectrum from 54 MHz to 550 MHz, digital signals transmitted downstream from a CMTS is modulated as analog signals and trans-

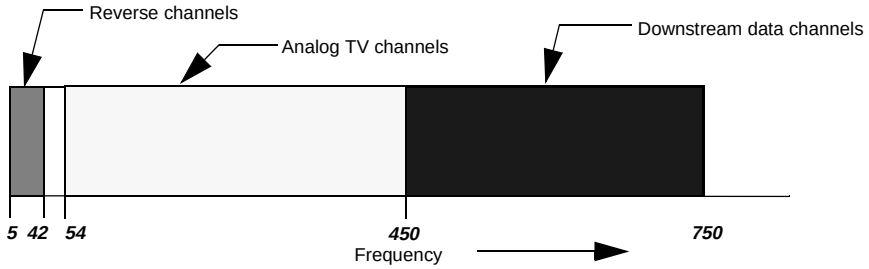


Figure 2. Frequency spectrum allocation for data services in an HFC network

mitted in one or more channels in the high frequency band - from 550 MHz to 750 MHz (see Figure 2). The number of channels used for data transmission over the HFC network is determined based on the subscriber base. For each downstream channel, the CMTS has a corresponding transmitter. To route information downstream over the HFC network to subscribers' homes, the CMTS maintains a mapping of subscriber cable modems to channels. Based on the destination addresses of the data packets, the CMTS decides to route packets over the appropriate HFC network channel. To ensure security, since an HFC network is essentially a broadcast network downstream from the headend, a CMTS performs encryption prior to transmission of packets downstream.

Upstream communications from a subscriber's home to the server complex trace back the downstream path. For each of the upstream channels, the CMTS has a corresponding receiver. The receivers operate in the low frequency band - 5 MHz to 42 MHz (see Figure 2). This low frequency band is highly susceptible to noise interference from electrical sources. For instance, unterminated cables in the home often act as antennas that pick up electromagnetic radiation from the surrounding environment, in the form of noise produced by home appliances, such as hair driers, mixers, vacuum cleaners, etc., and feed the noise back over the cable. Noise from different homes tends to aggregate close to the headend, reducing the noise margins and increasing the bit error rates of the channels, impacting the performance of data services. To overcome such ingress noise which is time and frequency dependent, the CMTS implements dynamic frequency selection. By monitoring error rates that it observes during its normal operation, the CMTS is capable of altering the channels used for data services.

Whereas the CMTS is the only transmitter on the downstream channels, on the upstream channels, multiple cable modems (CMs) of subscribers may contend for transmission. The CMTS takes responsibility for detecting and resolving contention between CMs for access to the channels. The upstream and downstream bandwidths available to subscribers are dependent upon the hardware capabilities in the CMTS and CMs, and the design of the CMTS-CM link. Our testbed network adopts an asymmetric approach, offering up to 10 Mbps downstream and about 700 Kbps upstream.

- ***Cable Plant:*** Fiber-optic links from the headend transport signals to a number of neighborhood nodes called *Fiber nodes*. These fiber nodes perform the optical-to-electrical signal conversion necessary to transport signals received from the headend over coaxial cables to subscribers' homes. A tree-and-branch structured coaxial cable network connects the fiber nodes to groups of 500-2000 homes. In the ideal architecture, by using independent fiber links

from the headend to each fiber node, an independent frequency band is allocated for each fiber node. In the initial stages of the deployment of HFC networks, cable operators may choose to save on fiber transmission and reception costs by reusing the same frequency band to serve multiple fiber nodes. In our testbed network, groups of 4 to 6 fiber nodes share the same frequency spectrum.

- ***Subscriber home equipment:*** In a subscriber's home, access to data services over the HFC network is enabled through a cable modem. The CM contains modulation-demodulation hardware to receive and transmit signals over the HFC network. To receive information from the CMTS, the CM tunes to the appropriate channel (instructions to enable a CM to find the frequency to tune to are specified as part of the CMTS-CM link protocol), demodulates the signal transmitted by the CMTS, receives the data that is addressed to it, and forwards this data to the subscriber's PC.

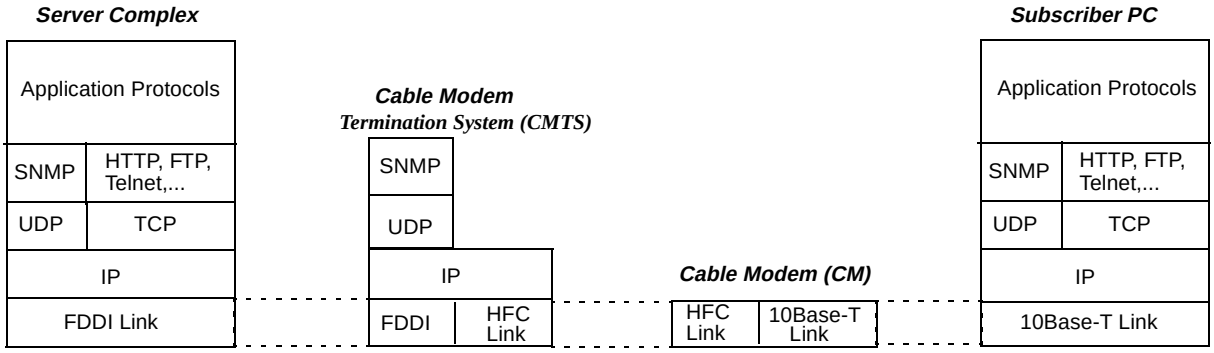


Figure 3. Protocol layering in the HFC network architecture

Figure 3 depicts the protocol layering in the HFC network architecture. The applications themselves are distributed: the client component executes on the subscriber's PC, communicating with server component(s) in the headend server complex. The standard Internet protocol suite (IP/UDP/TCP) is used for communication between the server complex and the Internet. IP packets are transported over FDDI between the server complex and the CMTS. IP packets encapsulated into HFC link packets are communicated over the HFC link.

3 Monitoring of Hybrid Fiber Coaxial Networks

The susceptibility of HFC networks to a variety of radio-frequency problems caused by factors such as exposure of the network to harsh environmental conditions, improper wiring and cable termination in subscriber homes, malfunctioning network equipment, improper balancing of signal levels, etc., has been well documented in literature [6]. Since subscribers generally tolerate a wide latitude in performance variations of television signals, the great majority of cable operators have not had to instrument their cable networks for automated, continuous monitoring. In most cable systems today, network faults are detected reactively, based on subscriber complaints [9]. Network engineering, to provide improved performance, is done manually and in an ad-hoc manner [1]. Problem diagnosis is usually enabled by expen-

sive special-purpose instrumentation such as spectrum analyzers, signal-level meters, and bit-error rate monitors that track the status of the physical network - the cables, amplifiers, fiber nodes, network taps, etc.

In [8], we had proposed the use of monitoring capabilities built into the data service equipment - the cable modems, CMTS, and subscriber PCs - as a low-cost instrumentation alternative that can enable proactive monitoring of HFC networks, so that problems can be detected and even possibly fixed before subscribers notice and complain. Such proactive monitoring is likely to be more crucial in the future because high availability and the capability to deliver on its promise of superior performance compared to more conventional ways of accessing data services are sure to be two of the essential requirements for subscriber acceptance and popularity of HFC networks.

In this paper, we present experimental results from monitoring a real-world HFC network demonstrating that even with minimal monitoring capabilities available in the first generation cable modems and CMTSs, it is possible to design sophisticated management applications to detect and report problems that occur in HFC networks proactively. Recently, Multimedia Cable Network Systems (MCNS), an organization co-founded by a number of leading cable operators, has defined a Data Over Cable Service Interface Specification (DOCSIS) intended to foster multi-vendor equipment compatibility. As part of this interoperability effort, MCNS is in the process of defining a comprehensive set of management interfaces for cable modems and CMTS equipment that is further likely to increase the utility of in-built instrumentation for managing HFC data networks.

3.1 Configuration of the Testbed Network

Our testbed network comprises of a dozen CMTSs serving different neighborhoods (each with 20,000 to 50,000 homes) in a US metropolis. An HFC neighborhood is served by a number of fiber nodes. As indicated in Section 2, the frequency spectrum is shared among four to six fiber nodes. The grouping of fiber nodes sharing the same spectrum is

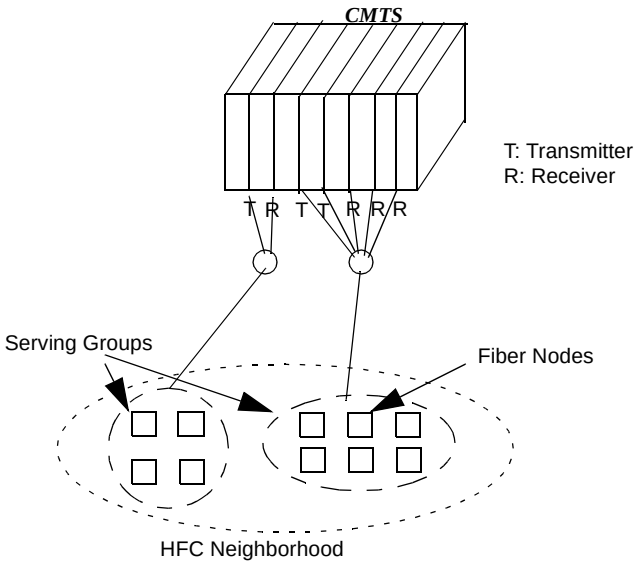


Figure 4. The testbed network configuration

referred to as *serving groups*. Downstream transmitters and upstream receivers of a CMTS are exclusively associated with a serving group. The number of transmitters and receivers associated with a serving group is determined based on the current and projected subscription rates in that serving group. Usage patterns of subscribers also affects the number of transmitters and receivers necessary. For instance, web surfers generally generate highly asymmetric traffic (downstream to upstream bandwidth usage ratios of 20:1 are common), whereas subscribers working from home generate symmetric traffic. Since upstream bandwidths are relatively much smaller compared to downstream bandwidths, typically, there are more receivers allocated to a serving group relative to the number of transmitters. Figure 4 depicts the mapping of fiber nodes to serving groups in an HFC neighborhood.

3.2 Management Support in the Testbed Network

The CMTS is the primary source of management information in the testbed network. Configuration information such as the number of transmitters and receivers in a CMTS, the serving groups associated with the CMTS, and the mapping of transmitters and receivers to serving groups is exposed via a proprietary Simple Network Management Protocol (SNMP) management information base (MIB). Using this interface, a management application can remotely configure the CMTS as well as query its configuration.

The CMTS MIB also exposes other information that the CMTS maintains for its normal operation. Status information such as the channels being currently used by the transmitters and receivers and the identities of the subscriber devices (cable modems and PCs) currently actively using the HFC network, performance information such as error rates and congestion levels on the upstream and downstream channels, and usage information such as packet transmission rates on the upstream and downstream channels, are all made available to management applications via the CMTS MIB.

In our testbed network, subscriber cable modems are not directly manageable. Instead, information about the CMs is made available via a proxy MIB that the CMTS supports. Using this MIB, a management application can track the rate of packets transmitted to and received by a CM.

3.3 Management Functions

There are three basic management roles that any HFC network monitoring solution must be targeted towards:

- *Network maintenance* relates to the proper operation of the HFC network in its existing configuration. An important task of network maintenance is proactive monitoring to detect trends in network performance (e.g., increase in packet error rate over time). Deterioration in performance of the HFC data network usually signifies the existence of RF impairments. Based on the results of proactive monitoring, the network maintenance function can schedule maintenance in advance of subscriber complaints. Since not all problems can be proactively handled (e.g., cable cuts, component breakdowns can happen suddenly), reactive management is also essential.
- *Subscriber support* relates to the handling of a subscriber complaint about the data service. Subscriber support is an end-to-end diagnosis process that attempts to determine whether the problem that a subscriber is experiencing is related to the Internet, to the on-line service providers, to the server complex, to the HFC network, or to the subscriber's PC. If the problem is determined to be caused in the HFC network, subscriber support is responsible for iso-

lating the problem (whether it is specific to one subscriber, or whether it has a more global scope). Typically, detailed problem diagnosis is delegated to the network maintenance function, which handles further troubleshooting.

- *Planning* handles the aspects of management that require reconfiguration of the network. A typical function of planning is network congestion management. To ensure subscriber satisfaction, the management system must be able to detect symptoms of overload on the transmission network, the CMTSs, and the servers, and then recommend measures for handling the overload: whether additional buffer memory should be added to the CMTSs, whether additional CMTS transmitters and receivers should be added, whether additional frequency spectrum should be allocated to data services (possibly at different times of the day), or even whether changes have to be made in the physical HFC network topology (such as partitioning of fiber nodes into smaller serving groups, so that more frequency spectrum is available per serving group).

In the next section, we describe a monitoring application, *hfcmon* that performs continuous, on-going monitoring of the tesbed network for the purposes of network maintenance and capacity planning. Subscriber support issues for cable networks are outside the scope of this paper.

3.4 Monitoring Application for the Testbed Network

Since many operational problems and faults that occur in an HFC network are topology related (e.g., ingress noise impacting one serving group of an HFC neighborhood), any monitoring application for an HFC network must be able to discover the topology of the network and correlate problems with the network topology. By utilizing the configuration aspects of the CMTS MIB, the monitoring application for our testbed network, *hfcmon* discovers the HFC network topology: the different neighborhoods serviced by the CMTSs, the serving groups in each neighborhood, and the mapping of transmitter/receiver cards to serving groups (see Figure 5). Since the HFC network may evolve over time (e.g., transmitter and receiver cards may be added to CMTSs with subscriber growth), *hfcmon* periodically polls the CMTSs to detect configuration changes.

Using the discovered HFC network topology, *hfcmon* proceeds to measure the quality of service (QoS) being delivered to subscribers, which is then made available to network maintenance personnel for scheduling appropriate repairs in the

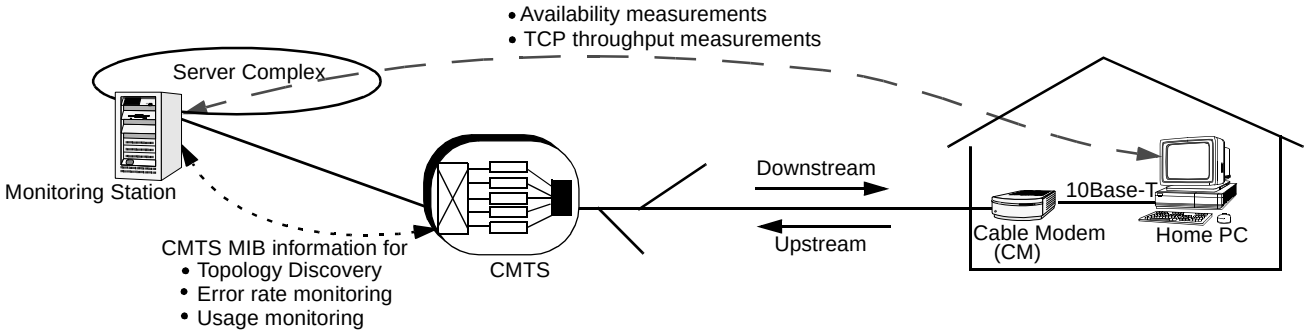


Figure 5. Design of *hfcmon*, an HFC network monitor. *hfcmon* uses CMTS MIB information for HFC network topology discovery and usage monitoring. QoS monitoring is based on active measurements to subscriber PC targets.

HFC network. In addition, *hfcmon* also provides usage information about the different CMTS transmitters and receivers, which can serve as the basis for capacity planning. The following sections discuss both of the above applications of *hfcmon*.

In our initial implementation, *hfcmon* is centralized - it executes on a monitoring station that is centrally located in the server complex. Future implementations are likely to employ distributed components, so as to scale to large HFC network deployments with hundreds of CMTSs.

4 Quality of Service Monitoring for HFC Networks

4.1 Enabling QoS Measurements

In the absence of adequate instrumentation in the application servers, in the CMTS, and cable modems, and in subscriber client applications, *hfcmon* uses active measurements that emulate typical subscriber activity to assess QoS. The potential for passive measurements, which utilize subscriber traffic itself for QoS measurements and thereby avoid the overheads of traffic generation for making the measurements, are discussed elsewhere [2].

In the absence of monitoring support in the CMTS and CMs, *hfcmon* targets measurements to subscriber PCs directly (other IP addressable devices, e.g., networked printers, connected to the CMs may also be used as targets). In the future, as the MCNS-standard CMs are expected to be IP addressable, the CMs themselves can be used as targets for *hfcmon*. The monitoring tools that *hfcmon* uses have been chosen so as to not require any special software support at subscriber PCs. The frequency of the measurements used by *hfcmon* is tuned to ensure that the active measurements themselves do not impact subscriber perceived QoS.

Since *hfcmon* operates from a monitoring station in the server complex, measurements from the monitoring station to a subscriber PC traverse the network link between the monitoring station to a CMTS, in addition to the HFC network between the CMTS and the subscriber's CM and the local network connecting the subscriber's PC to the CM. To deduce the state of the HFC access network in isolation, *hfcmon* also simultaneously assesses the availability and performance of the network links between the server complex and the CMTSs. By correlating the end-to-end (i.e., monitoring station to subscriber PC) measurements with the monitoring station to CMTS link measurements, *hfcmon* estimates the QoS delivered over the HFC access network. In the measurements described here, the links between the server complex and the CMTSs were never a bottleneck. A random subscriber PC selection scheme described later is used to ensure that *hfcmon*'s results are not biased by bottlenecks in a subscriber home (either in the local network, or in the subscriber's PC). Together these constraints ensure that the end-to-end measurements directly provide an estimate of the HFC network status.

Although different subscribers may have different perceptions of QoS, measuring the QoS individually for each subscriber is overly expensive. Furthermore, such monitoring is usually unnecessary since a great majority of HFC network outages (e.g., cable cuts, amplifier failures, ingress noise impact) affect a large group (a few thousand to several thousand) of subscriber homes. Since the topology discovery performed by *hfcmon* only provides it with information about the HFC neighborhoods and their serving groups, but not the individual fiber nodes that make up a serving group,

or the various branches that feed off a fiber node, *hfcmon* is only able to assess and report problems for the different serving groups. Additional diagnostic measurements are necessary to further diagnose problems to specific fiber nodes or branches in the HFC network.

4.2 QoS Metrics and their Monitoring Tools

Subscribers perceive quality of service of data services in terms of the availability and performance of the services. Whereas availability of data services refers to whether a subscriber is able to connect via his/her cable modem to servers in the server complex, performance of data services refers to how good the connection is when the connection was established. As they relate to the HFC access network, availability and performance are defined and measured as follows:

- *Availability*: Availability is defined as the percentage of time that a subscriber's PC via a CM is able to maintain a communication channel for packet transmission to and from a CMTS. If a target PC exists, availability can be measured by sending Internet Control Message Protocol (ICMP) Echo packets from the monitoring station to the PC (e.g., using the *ping* utility), which in turn solicits an ICMP Echo response back from the PC. If such a response is received, *hfcmon* assesses the serving group in which the target PC exists to be available. As we will see later, the packet loss and round-trip delay values measured during this test can also serve as useful diagnostic measurements for performance problems that may occur in the HFC network.

  To be able to measure availability of each of the serving groups in the HFC network with sufficient confidence, *hfcmon* requires targets in each serving group that are available at all times. Since subscribers' PCs are not guaranteed to be available for testing at all times, to assess availability, cable operators must deploy dedicated targets, one for each serving group. As an experiment, in our testbed network, we have used "*test cable modems*", which are dedicated cable modems installed in a few of the serving groups of the HFC network for the availability measurements. HP JetDirect print server boxes are used as low-cost IP addressable devices connected to the dedicated cable modems that serve as targets for the availability measurements. A side benefit of our deployment of the JetDirect servers is that since they support standard SNMP MIB-II interfaces, they enable additional diagnostic measurements to determine the relative magnitudes of packet loss in the downstream and upstream directions independently, using the methodology described in [8].

- *Performance*: A majority of applications being initially considered for broadband services are data transfer applications (e.g., Web, E-mail, News) that utilize the TCP protocol for reliable communication. Moreover, all of these applications are primarily asymmetric, mainly involving data retrievals from a server to a subscriber's PC. To assess subscriber-perceived performance of the HFC network, *hfcmon* uses throughput, defined as the rate of data that can be transferred *reliably* over the HFC network from a server to a subscriber's PC, as a performance metric.

  As per the above definition, since reliable transmission is assumed, throughput is measured above the TCP layer in the protocol stack. Although a number of tools such as *netperf* and *throughput TCP (ttcp)* exist for throughput monitoring, all of these tools require special software agents at subscriber PCs for this purpose. We found the public domain *traceroute Reno (Treno)* tool to be an attractive alternative since it does not require special client-side soft-

ware support [7]. *Treno* emulates a TCP-based data transfer using User Datagram Protocol (UDP) packets to transmit data to a target. The UDP packets are sized equivalent to typical TCP packets and are directed at a non-existent port on the target. Any IP addressable target responds to such packets by transmitting an ICMP error message that is almost equivalent in size to the TCP acknowledgment packets that a subscriber PC transmits in response to a TCP data packet. By incorporating TCP's delayed acknowledgment, window opening, and retransmission schemes in the server application, *Treno* emulates data transfers over a TCP connection.

In its basic form, *Treno* was intended to permit subscribers to compare the performance offered by different network providers, and for network providers to monitor the performance of their networks. In an effort to measure the best possible throughput achievable when using TCP as the transport protocol over an IP network, *Treno* continues to open its transmission window (emulating TCP's slow-start) until it encounters a packet loss. This could potentially result in flooding of the network for the duration of the test. Moreover, in practice, the TCP stacks on subscriber PCs are configured with a default maximum window size that restricts the amount of data transmitted simultaneously. Since it has no window size restriction built in, *Treno* over-estimates the throughput achieved by a subscriber during a normal TCP-based data transmission. To overcome this limitation, *hfcmon* extends the concepts of *Treno* by building in restrictions on the maximum window size. Furthermore, by restricting the amount of data transferred during each measurement to match typical data transfer sizes that subscribers use when retrieving Web, Email, and News content, *hfcmon* ensures that the measurements reflect subscriber perceptions of network throughput.

4.3 Implementation of QoS Measurements

For the serving groups with dedicated "test cable modems", *hfcmon* monitors availability by simply targeting the corresponding test cable modems. The list of pre-configured test cable modems is provided to *hfcmon* as part of its configuration specifications. Availability alerts are issued to a network management platform such as HP OpenView, whenever lack of connectivity is detected to any of the test cable modems. For a serving group that is not associated with a dedicated target, *hfcmon* cannot assess the availability of the serving group directly. Instead, *hfcmon* can only identify the times when no subscriber CMs in that serving group are on-line and those times when at least one subscriber CM in that serving group is on-line. There are several ways of determining the number of subscriber CMs that are on-line:

- *Using the CMTS MIB*: When a CM in a subscriber's home is powered on, the CM has to first communicate with the CMTS to determine the channel(s) to tune to, to set the transmission and reception signal levels, and to obtain the appropriate security keys to use for encryption/decryption of messages. During its normal operation, a CMTS may keep track of the number and identity of CMs that are on-line. Heart-beats implemented at the data link layer may be used by the CMTS to track the CMs that are on-line at any time. This information can then be exposed via the CMTS MIB.

  However, this capability does not exist in our testbed network, since the CMTS only tracks the number of CMs that are actively transmitting or receiving packets at any time, not all the CMs that are on-line (i.e., CMs that are on-line but not active are not tracked by the CMTS).

- *Scanning the available IP address space*: In the absence of support from the CMTS, our implementation of *hfcmon* relies on IP-level mechanisms to detect subscriber PCs (and the corresponding CMs) that are on-line. In our testbed network, each HFC neighborhood is comprised of two Class C IP sub-networks, each with 256 possible IP addresses. Since the range of possible IP addresses is limited, *hfcmon* attempts to send an ICMP echo packet to each of the possible IP addresses. Only PCs that are on-line respond to these packets. As it forwards responses from these PCs to the server complex, the CMTS alters its internal tables to reflect that the PCs that responded and their corresponding CMs are active. By querying the CMTS MIB, *hfcmon* can then determine (i) the CMs associated with PCs, (ii) the association of CMs to serving groups (since this information does not change often, to minimize SNMP overheads at the CMTSs, *hfcmon* maintains an in-memory cache of this information), and (iii) the transmission and reception channels used by the CMs. This information is used not only for the availability measurements but as we will see later, for the performance measurements as well.

- *Exploiting monitoring support in the subscriber management system*: The above approach of scanning the entire address space to detect subscribers and their PCs/CMs is not suitable for deployment in networks with hundreds of thousands of subscribers, since the overheads of such active measurements are excessive. A more scalable solution is to define queryable interfaces to the subscriber management system. Using these interfaces, a monitoring application such as *hfcmon* can discover the IP addresses of subscriber PCs that are on-line at any time. Using the CMTS MIB, *hfcmon* can then determine the association between subscribers and serving groups.

In the absence of dedicated targets that it can use, *hfcmon* relies on choosing one or more subscriber PCs that are on-line for performance measurements. Even when there are dedicated targets in the network, *hfcmon* may prefer to choose a different target for the measurement in an attempt not to be biased by a specific target location (which may either be very high performing or very poor performing). Once a list of subscriber PCs and CMs that are on-line is available and their mapping to the different serving groups is determined, *hfcmon* chooses one target for each serving group. For each target, *hfcmon* uses ICMP Echo (i.e., the *ping* utility) to estimate the packet loss and round-trip delay during transmission of a series of packets to this target. This is followed by a throughput test to the subscriber PC. The data transfer size is chosen to reflect typical Web, Email, and News transfers. Since the PCs and CMs that are on-line vary from time-to-time, *hfcmon* periodically repeats the above procedure, selecting a target each time according to the availability of the targets and a test policy that is described in Section 4.4.

In order to provide detailed diagnosis of problems when they happen, *hfcmon* also tracks other information available from the CMTS MIB. For instance, by tracking the channels in use for the transmitters and receivers associated with a serving group, *hfcmon* can detect instances when the CMTS decides to change the channels in use for that serving group (possibly because the CMTS has detected unusually high noise levels in the channels). When a performance problem is detected by *hfcmon*, a network operator has to correlate the channel changes made by the CMTS with the QoS measures of *hfcmon* to determine whether the CMTS is not functioning properly (i.e., it is not dynamically moving to a more usable channel even though subscriber-perceived performance has degraded) or whether the HFC network needs immediate maintenance (i.e., all available channels have degraded).

4.4 Implementation Experience

The measurements described in the previous section have been deployed for several months in our testbed network and have been proven to be remarkably effective in detecting potential HFC network problems well in advance of subscriber complaints. This section presents real-world examples illustrating the practical utility of *hfcmon*.

As mentioned earlier, in our testbed network, *hfcmon* is executed from a monitoring station in the server complex. HFC network availability and performance measurements are made every 5 mins, and a customized Excel user interface enables easy visualization of the measurement results as they become available on the monitoring station. To perform its tasks, *hfcmon* relies on the availability of a minimal set of SNMP capabilities (*snmpget* and *snmpwalk* utilities) on the monitoring station. In our testbed network, the round-trip propagation time between the server complex and subscriber homes is estimated to be about 40ms. For a TCP window size of 8Kbytes and a data transfer size of 50Kbytes, the round-trip propagation time restricts the observed throughput to a best-in-class PC to about 2Mbits/sec. Depending on the processing capabilities of subscriber PCs, throughput varies significantly. Experiments in our testbed network indicated that a great majority of subscriber PCs are capable of sustaining a throughput of over 500 Kbits/sec. To ensure that subscribers perceived the cable data service to be at least as good as a telephone company's ISDN access service, the cable operator targeted to provide a minimum throughput of 128 Kbits/sec (ISDN speed).

4.4.1 Performance Problem Identification

The utility of the throughput measurements made throughout the day by *hfcmon* to different HFC serving groups is demonstrated by Figure 6. This figure depicts the summary of performance observed to the different serving groups of an HFC neighborhood served by one CMTS. By comparing the performance of the different serving groups, a network

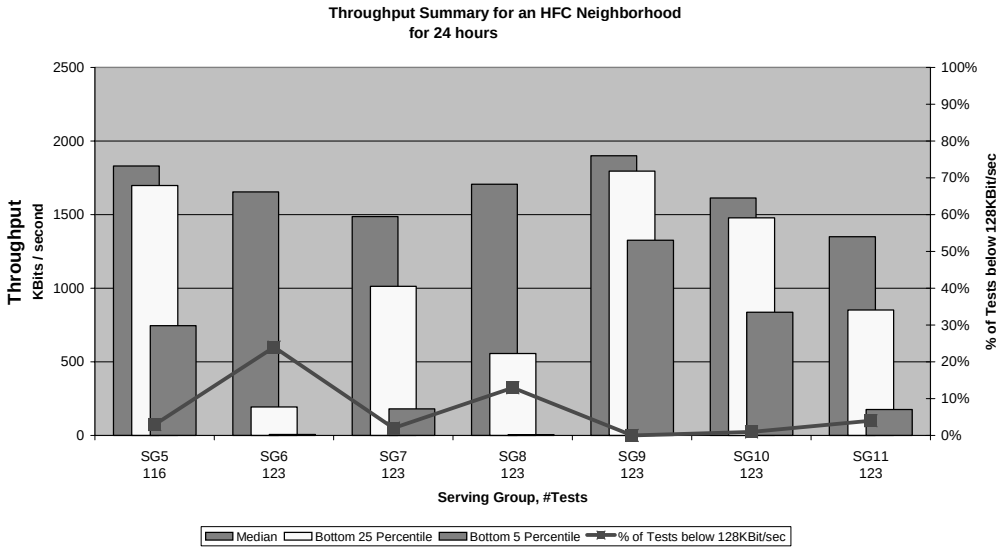


Figure 6. Summary of performance observed to an HFC neighborhood during a 24 hour period. The X-axis indicates the different serving groups in the neighborhood and the number of tests made to each serving group during the 24 hour period. Notice from the percentage of tests that do not meet the 128 Kbps threshold that serving groups SG6 and SG8 are performing much worse than the others.

operator can quickly determine serving groups that are experiencing performance problems. The vertical bars in Figure 6 provide an idea of the throughput distribution over time. While the median value is an indicator of how good a subscriber's connection usually is, the 5 percentile value is an indicator of how bad the performance can get sometimes. To provide a instant indicator of the relative performance of the different serving groups, the percentage of measurements for which the measured throughput is below the 128 Kbits/sec threshold is also depicted in the figure by the solid line which is plotted against the axis on the right. For serving groups that do not require any maintenance (e.g., SG9), almost all the tests exceed the threshold. Based on experience, we have observed that serving groups that do not meet the threshold throughput criteria at least 10% of the time (e.g., SG6 and SG8 in Figure 6) are in need of urgent maintenance. The 5 percentile throughput values in Figure 6 also provide an idea of the relative states of degradation of the different serving groups. For example, although serving groups SG7 and SG11 meet the 128 Kbps threshold almost all the time, the 5 percentile values for these two serving groups is much worse than those for the serving groups SG5, SG9, and SG10.

While in a great majority of cases, there is very little correlation between problems affecting different serving groups, in a few cases, the problems may be related. Figure 7 depicts a case when all serving groups of an HFC neighborhood are being affected. Since the different serving groups use distinct portions of the HFC network, and since they use independent frequency spectra, problems such as the one in Figure 7 that affect all the serving groups are likely to be caused closer to or in the network headend (e.g., in the signal splitters and combiners, multiplexing equipment, or in the CMTS itself).

4.4.2 Performance Problem Diagnosis

There may be several reasons why a serving group may not be performing to expectation. The performance summary depicted in Figures 6 and 7 is not sufficient to indicate the nature of the problems affecting each serving group. For fur-

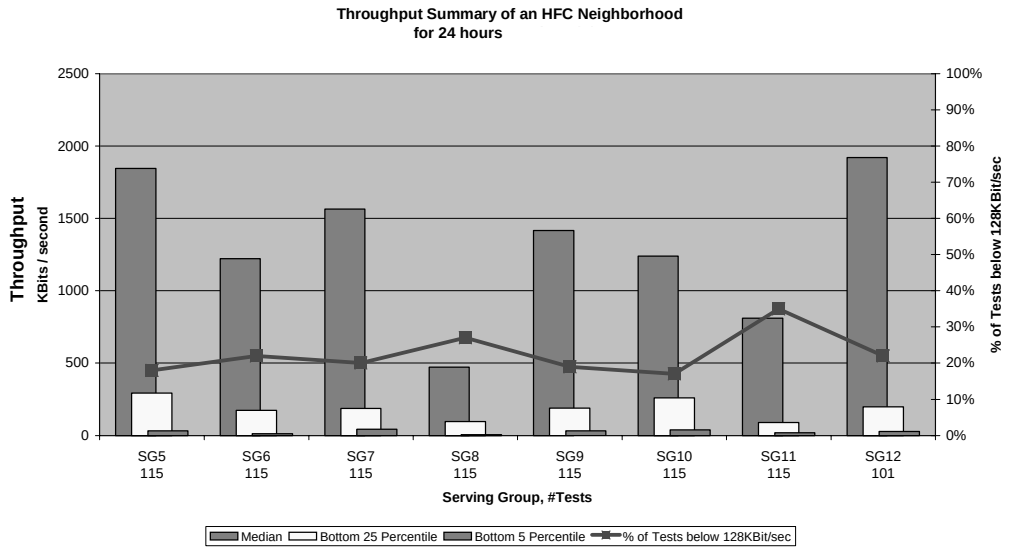
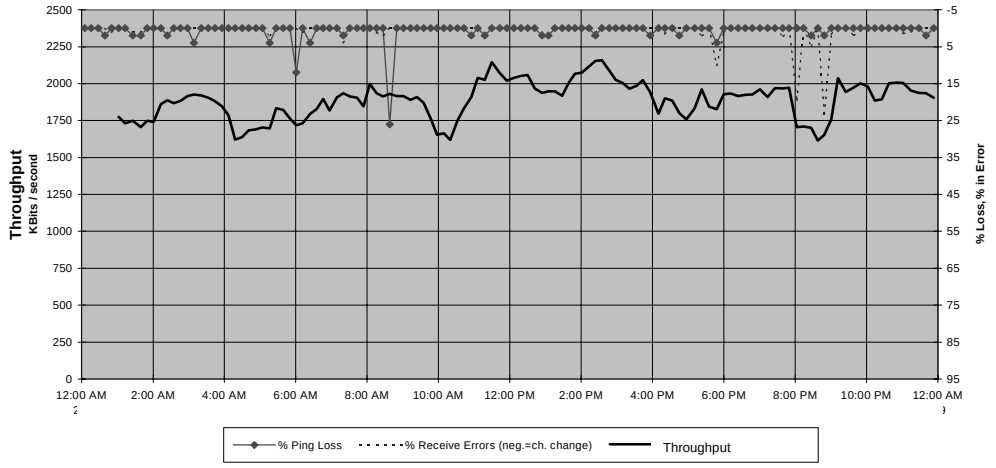


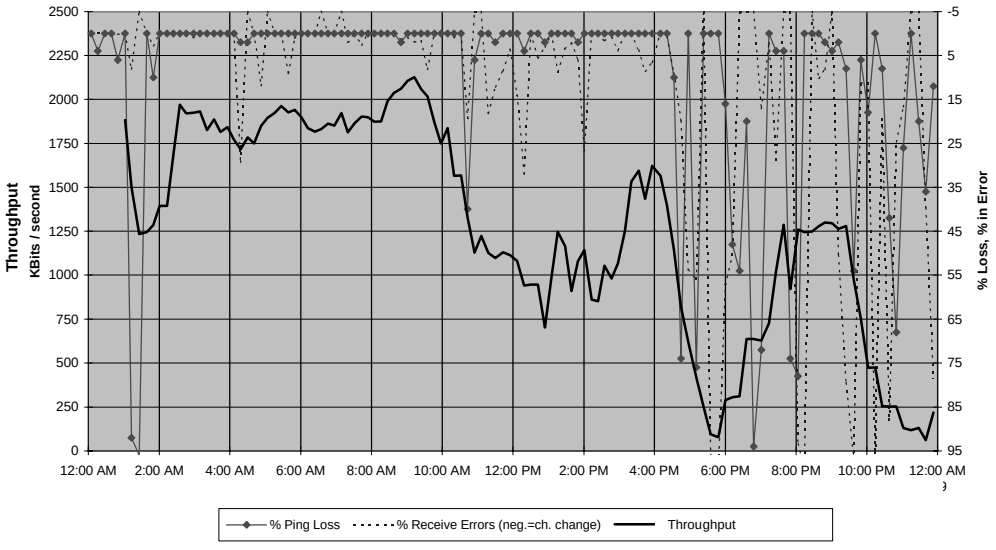
Figure 7. A problem scenario impacting performance to all serving groups of an HFC neighborhood

ther diagnosis, more detailed performance information which describes the variation of throughput with time of day is necessary. Figure 8 contrasts the performance of a serving group that is operating normally with that of a serving group that is experiencing problems. In the former case, the moving average of throughput is fairly constant, whereas in the latter case, there is a significant drop in throughput at several times of the day. The graph of throughput variation over time indicates the exact time periods when the cable operator must schedule network maintenance (e.g., after 4pm in the example in Figure 8(b)).

Typically, a majority of HFC network problems are caused by ingress noise problems that cause bit errors during transmission, resulting in packet loss seen by the network applications. As indicated in [3], even a relatively small packet



(a)



(b)

Figure 8. (a) depicts the normal performance of an HFC serving group; (b) Performance of an HFC serving group that is experiencing packet loss problems

loss rate (1-5%) can result in a significant drop (of 10-50%) in throughput. In addition to tracking throughput variations, Figure 8 also tracks the packet loss variation with time of day. Notice in Figure 8(b) that the five to ten-fold drop in throughput is accompanied by significant packet loss. Figure 8(b) also illustrates that during the same time-periods, the CMTS detects a significant percentage of errors (indicated by the dotted line in the figure) during signal receptions on the upstream channels and frequently hops channels in a bid to improve the performance delivered to subscribers (In Figure 8, negative values of percentage errors are used to indicate times when the CMTS receiver changes channels).

In a few cases, throughput reduction may not be accompanied by packet loss. An increase in round-trip delay of packets, since it slows down TCP's flow control algorithms can also result in throughput reductions. Such an increase in round-trip delay can be caused either by an over-utilization of the network (e.g., congested downstream channel), or by malfunctioning network equipment (e.g., CMTS). To detect such problems, *hfcmon* tracks the minimum, average, and maximum round-trip delays experienced by packets during the course of the availability measurements. During our experiments, we observed that since subscriber PCs are used as targets, the round-trip delay measurements are particularly susceptible to the state of the subscriber PC targets. For instance, certain TCP/IP stacks employed in the subscriber PCs treat ICMP echo packets (used for the delay measurements by *hfcmon*) at low priority when subscribers are actively accessing the network. Consequently, delays of over a second are commonly seen to such subscriber PC targets and are not to be construed as indicators of HFC network problems. Based on these observations, we use the minimum round-trip delay (rather than the average or maximum values) as an estimate of the HFC network performance.

4.4.3 Measurement Policies for *hfcmon*

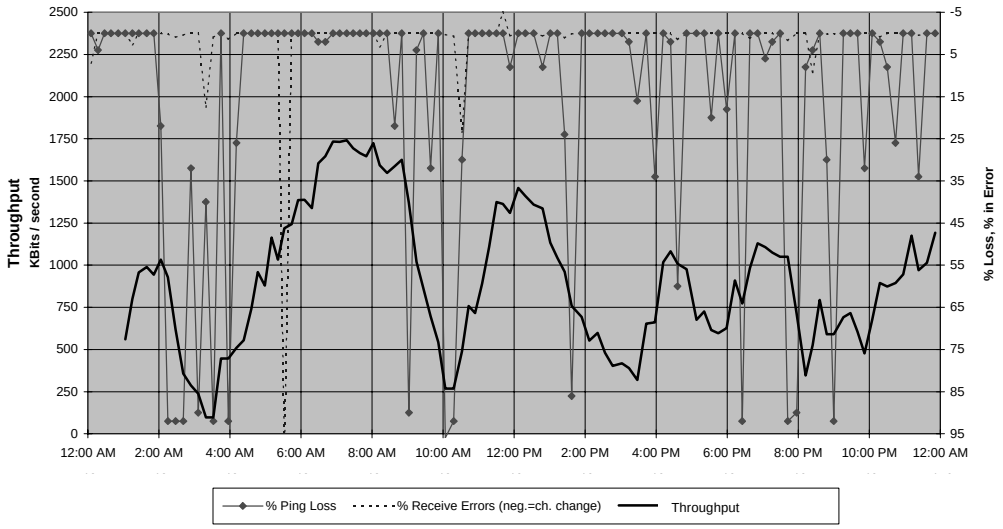
When assessing the performance of a serving group, *hfcmon* can employ different policies for choosing the target for each measurement. We have experimented with two policies:

- In a minimal change approach, while monitoring a specific serving group, *hfcmon* makes an initial random choice of the target for that serving group. If the target achieves at least the threshold value of throughput during the first measurement, in subsequent measurement periods, *hfcmon* continues to use the same target until the time when the target is no longer available for testing (because it may have been powered off in a subscriber's home). When this happens, *hfcmon* then picks another target at random from among the targets currently on-line for the serving group under consideration.
- In an alternative, maximal change approach, during each measurement period, *hfcmon* chooses a target at random from among all the targets that are on-line for a serving group. By doing so, *hfcmon* strives to obtain a better approximation of the quality of service being provided to subscribers in the serving group.

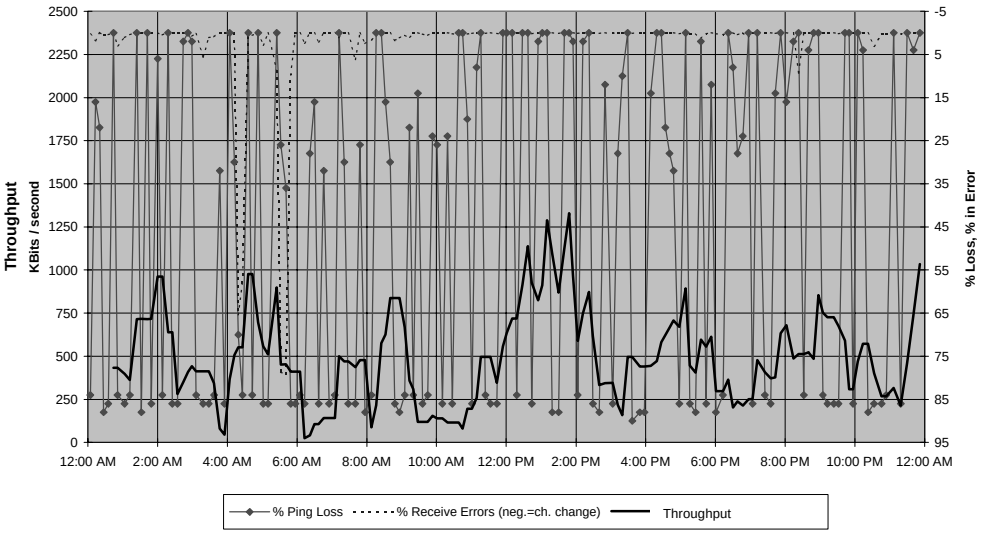
There are interesting trade-offs with both approaches. The minimal change approach attempts to monitor the performance delivered to the same target for a long time, and thereby provides a better indication of changes in performance over time. On the other hand, since it monitors the same target each time, this policy may not detect problems if the target being considered is not experiencing problems (say because it is located closest to the headend), but all other devices in the same serving group are affected. In contrast, since it samples a number of targets, the maximal change approach has potentially a better chance of detecting problems that are affecting several subscribers. At the same time,

since it tests different targets (which could potentially have different processing capabilities) each time, this approach can demonstrate variations in throughput that may not be of much significance, since they may only reflect the heterogeneity of the targets. Hence, when using the maximal approach, it is necessary to consider the long-term trend in throughput, rather than the instantaneous values.

To evaluate the minimal and maximal change policies, we simultaneously executed two invocations of *hfcmon* using these policies on the same HFC network neighborhood. Since a great majority of HFC network problems are likely to affect all subscriber homes in a serving group, we had expected to observe little difference between the results of the two policies. To our great surprise, this turned out not to be the case (see Figure 9). Whereas the maximal change policy



(a)



(b)

Figure 9. Comparison of different implementation policies for *hfcmon*: (a) shows the results when using a minimal change policy; (b) shows the results for a maximal change policy

reported performance problems on two of the eight serving groups of an HFC neighborhood, the minimal change policy did not observe problems to the same extent. For instance, whereas the minimal change policy (Figure 9(a)) showed that the average throughput of a serving group is more than 500Kbps for most of the day, the maximal change policy (Figure 9(b)) indicated an average throughput well below 500 Kbps for the same serving group, for the same time period. Further testing revealed that problems indicated by the maximal change policy did indeed exist on two of the eight serving groups. While a majority of subscriber homes in a serving group were observing the problem, there were a few isolated homes that were observing normal performance. The minimal change policy happened to choose one of the latter group of homes as target and continued to use the same target throughout. In contrast, by constantly changing targets, the maximal change policy detected the existing problem. In this example, the problem turned out to be a defect in the CMTS implementation that caused some CMs to move into and remain in a low priority servicing state, wherein they received substantially poorer service than other CMs. Based on this experience, we have chosen to use the maximal change policy for *hfcmon*.

Since the maximal change policy requires access to the CMTS MIB each time it executes in order to discover a new target each time, this policy can impose a significant SNMP traffic load from the CMTS, thereby impacting its performance. To overcome this drawback, our implementation of *hfcmon* implements an intelligent MIB caching policy that minimizes the demands made on the CMTS.

5 Usage Monitoring for HFC Networks

Whereas monitoring of QoS measures such as availability and performance is critical for network maintenance, monitoring of the usage of the HFC network is critical for capacity planning. Our monitoring application, *hfcmon*, uses the

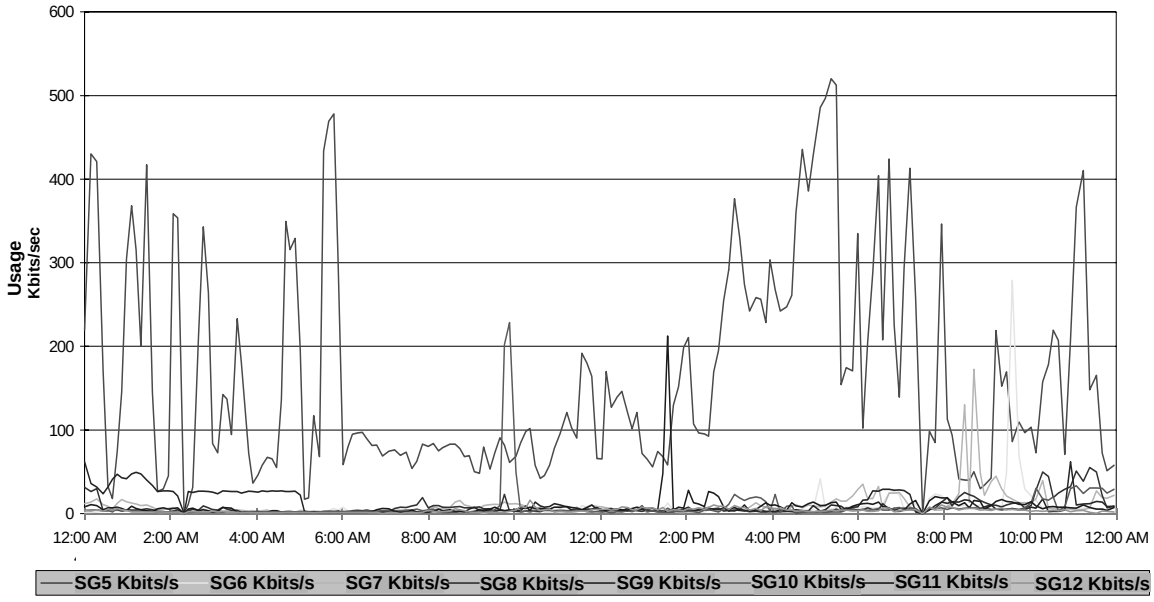
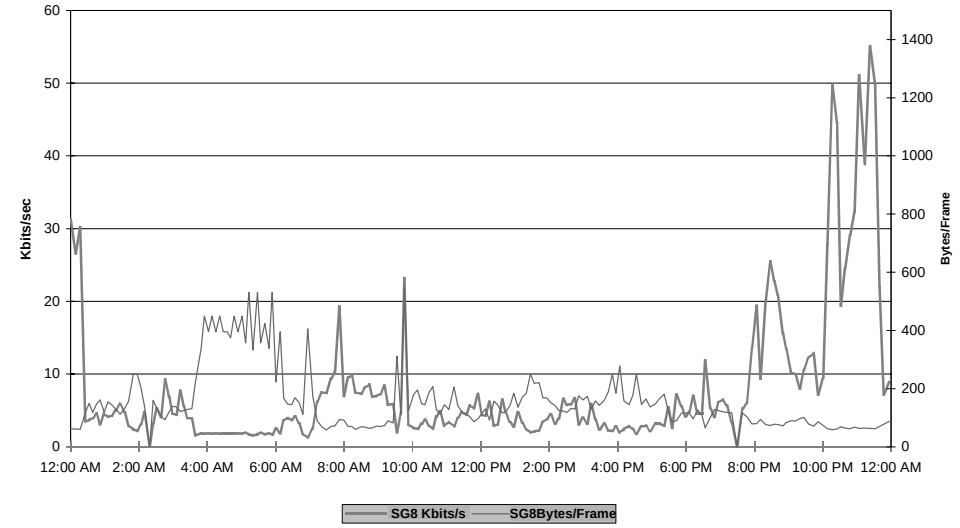
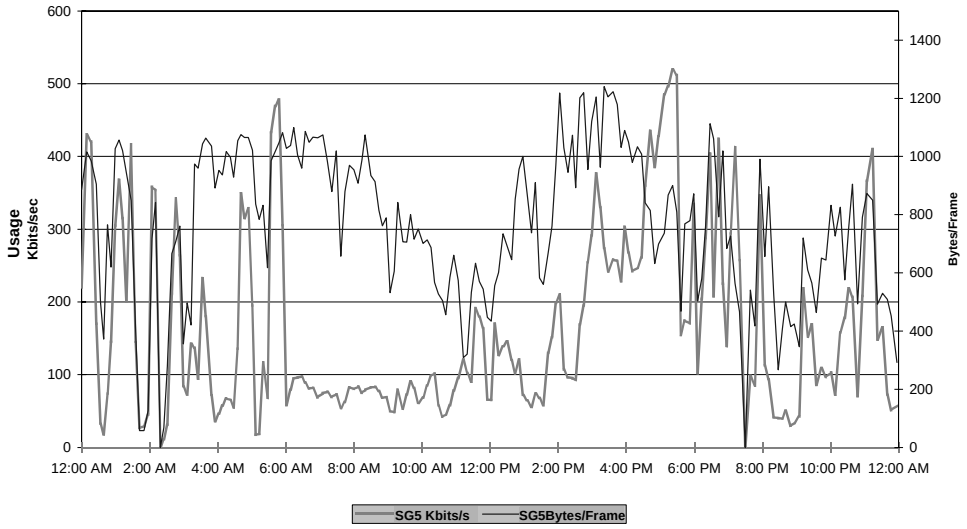


Figure 10. Comparison of usage of upstream channels of different serving groups in an HFC neighborhood

CMTS MIB to track usage of the different serving groups. By polling the CMTS MIB periodically, *hfcmon* computes the packet transportation rate over the upstream and downstream channels of a serving group. By providing the usage statistics for each of the receivers on a serving group, *hfcmon* permits load balancing capabilities of the CMTS implementation to be critically analyzed by a human operator. More importantly, the aggregate usage of the upstream and downstream channels can be used to determine whether additional transmitter or receiver cards need to be configured in the CMTS, whether additional spectrum needs to be allocated in the different serving groups, or whether the HFC network is being used in ways not anticipated by the cable operator.



(a)



(b)

Figure 11. Comparison of packet size distributions of traffic on two different serving groups: (a) SG8 and (b) SG5. SG5 which is experiencing a higher utilization on the upstream channels is seeing larger packets being transmitted upstream.

Figure 10 compares the usage of upstream channels in different serving groups of an HFC neighborhood during a 24 hour period. Each of the serving groups in this example is allocated one upstream channel with a maximum capacity of about 700 Kbps. Figure 10 illustrates a scenario in which usage of one of the serving groups (SG5) is hundred-times as much as the usage of the other serving groups. Since usage of SG5 during the peak hours is more than 50% of the theoretical maximum capacity of the upstream channel and since the efficiency of most upstream channel contention resolution protocols is known to be well below 100%, we conjecture that the upstream channel in use for SG5 is close to capacity during certain times of the day.

To enable further diagnosis of problems of increased usage, using the CMTS MIB, *hfcmon* tracks the average size of link-layer frames being transmitted on the upstream and downstream channels in the different serving groups. Since the great majority of applications enabled for subscriber access are asymmetric applications involving larger data transfers from the server complex to subscriber PCs (downstream), we would expect to see larger packets (TCP data packets of about 1Kbytes) on the downstream channels and much smaller packets (TCP acknowledgments that are usually less than 100 bytes in size) on the upstream channels. Figure 11(a) illustrates the expected pattern of traffic on the upstream channel of serving group SG8. Notice that even when the traffic load picks up after 8pm, the packet size distribution stays low. This is consistent with our expectation.

Surprisingly, on the upstream channel of serving group SG5, the average packet size is 600-1000 bytes (see Figure 11(b)). This suggests that a significant number of upstream data transfers from subscriber PCs to the server complex and beyond are occurring, which is an abnormality in the HFC testbed being considered. Further analysis revealed that the cause of the increased load was one of the subscribers on SG5, who had an operational web server on his PC that was receiving a high frequency of hits from the Internet. Using information such as this provided by *hfcmon*, a cable operator can detect and charge differentially for such instances of abnormal network usage, or at least plan for appropriate capacity improvements.

Figure 11 also quantifies the overhead of *hfcmon* in terms of network packet traffic generated for making the QoS and usage measurements. Since *hfcmon* is executing throughout the day, generating traffic at the same rate all the time, the traffic rate generated on the upstream channels by *hfcmon* is no greater than the minimum of the packet rate distribution curve. From Figure 11(a), we estimate the overhead of *hfcmon* on the upstream channels to be about 4 Kbits/sec on the average, which is only about 0.5% of the channel capacity. The average overheads on the downstream channels was computed to be about 50 Kbits/sec on the average, which amounts to about 1.6% of the link capacity.

6 Summary

In this paper, we have described experiences from monitoring a broadband hybrid fiber-coaxial access network deployment. Using real-world examples, we have illustrated that even with minimal monitoring capabilities built-into the HFC data networking equipment, it is possible to effectively monitor the status and performance of the HFC network. Since they rely on built-in instrumentation, our monitoring tools can enable low cost, yet effective management of HFC networks. Since many other emerging broadband access technologies such as Asymmetric Digital Subscriber Line

(ADSL) and Wireless Local Multipoint Distribution Service (LMDS) share many of the characteristics of HFC networks, our monitoring tools are applicable to these other local loop technologies as well.

References

[1] J. C. Anderson and P. Eng. An integrated network management system for cable television. In *Proceedings of the 43rd Annual NCTA Convention and Exposition, New Orleans, Louisiana*, pages 74–84, May 1994.

[2] M. Asawa. Measuring and analyzing service levels: A scalable passive approach. *Hewlett-Packard Laboratories Technical Report*, October 1997.

[3] R. Cohen and S. Ramanathan. Tuning TCP for high performance in hybrid fiber coax networks. *To appear in IEEE/ACM Transactions on Networking*, February 1998.

[4] T. Filanowski. QAM for broadband networks: The right choice. *Communications Technology*, pages 36,82–85, January 1995.

[5] D. T. Gall. Digital modulation on coaxial/fiber hybrid systems. *Communications Technology*, pages 42–45, January 1995.

[6] N. Himayat, C. A. Eldering, M. Kolber, and E. L. Dickinson. Characteristics of hybrid fiber-coax return systems. In *Proceedings of the SCTE 1995 Conference on Emerging Technologies, Orlando, Florida*, pages 191–200, January 1995.

[7] M. Mathis and J. Mahdavi. Diagnosing Internet congestion with a transport layer performance tool. In *Proceedings of INET'96, URL: http://www.psc.edu/ mathis/htmlpapers/inet96.treno.html*, June 1996.

[8] E. Perry and S. Ramanathan. Network management for residential broadband interactive data services. *IEEE Communications Magazine, Special Issue on Broadband Access*, Vol. 34, No. 11, pp. 114-121, November 1996.

[9] R. Schwarz and S. Kreutzer. Recipe for a successful OSS: Broadband networks demand operations support. *Communications Engineering Digest*, pages 92–100, June 1995.