Turkish Journal of Electrical Engineering and Computer Sciences

Volume 10 | Number 1 Article 3

1-1-2002

A New Congestion Control Mechanism Proposed for the SSCOP Protocol

A. HALİM ZAİM

Turk J Elec Engin, VOL.10, NO.1 2002, © TÜBİTAK

A New Congestion Control Mechanism Proposed for the SSCOP Protocol

A. Halim ZAİM
Department of Electrical and Computer Engineering,
NCSU, Raleigh, NC-USA ahzaim@eos.ncsu.edu

Abstract

International Telecommunication Union, Telecommunications Sector (ITU-T) Recommendation Q.2110 states that the update of a granted credit window to the transmitter (VR(MR)) is implementation dependent. In this paper, a new algorithm is designed to control the traffic flow coming from malfunctioning users. The VR(MR) parameter is controlled for that purpose. The simulation results show that this new technique nearly doubles the system performance.

Key Words: *Protocol Design, ADSL, SSCOP.*

1. Introduction

It is typical in the design of telecommunication equipment to distribute link (layer 2) protocols to the interface cards. This provides two advantages. First, layer 2 protocols typically involve keep-alive messages, which are sent continuously. These can be handled autonomously by the interface card without interrupting higher-level control units. Second, since interface card requests for uplink of layer 3 protocols are usually serviced in a fair share fashion, "noisy" users exhaust their buffers on the interface card and do not affect other users. For a protocol like the Service Specific Connection Oriented Protocol (SSCOP) [1], this arrangement is ideal. SSCOP has a dynamically sized transmit window. The transmit window is initially set to the size of the buffer on the interface card and incremented each time the interface card uplinks a message to the control unit. A disadvantage is that memory is not shared between cards and the statistical independence of layer 3 message arrivals cannot be used to share memory buffers.

During the recent design of a Switched Virtual Circuit (SVC) Asynchronous Transfer Mode (ATM) feature on an Asymmetric Digital Subscriber Line (ADSL) Access Network, the author was presented with the requirement that the SSCOP layer be centralized on the control unit. All ATM cells on the signaling channel would be passed transparently by the interface card. It was decided that the SSCOP buffer space would be shared by all (577) SSCOP connections to achieve an efficient use of Random Access Memory (RAM). Obviously, users sending too many layer 3 Protocol Data Units (PDUs) must be throttled. This paper presents the throttling mechanism and a simulation of the access network node under load.

This paper presents a congestion control mechanism for the signaling connections only. Congestion control on bearer connections is beyond the scope of this paper.

The paper is organized in the following manner: in section II, the architecture of an Access Network is explained. Afterwards, section III deals with the SSCOP flow control mechanism. In section IV, mathematical formulations necessary to calculate the window size are given. Section V focuses on the buffer usage during normal and abnormal traffic situations. After explaining the new throttling mechanism designed for isolating malfunctioning users in section VI, in section VII simulation results are given to express the performance improvements obtained with that new throttling mechanism. Conclusions are given in section VIII.

2. Access Network Architecture

In Figure 1, a bidirectional end-to-end ATM connection using ADSL Access Network is shown. As seen in the figure, the Access Network is connected to the ATM switch by means of a Synchronous Digital Hierarchy (SDH) Synchronous Transport Module 1 (STM-1) link. The Access Network contains an SDH Network Termination and a number of ADSL Line Interface Modules (LIM). The LIM contains the ADSL Line Terminal (LT) and the Plain Old Telephone Service (POTS) splitter functionality. The POTS splitter is connected to the Public Switched Telephone Network (PSTN) to allow the use of the POTS service. The POTS service and the ADSL channels are Frequency Division Multiplexed on the twisted pair towards the subscriber's home.

The connection between the Set Top Box (STB) / Personal Computer (PC) and the Access Network Transport (ANT) uses the 25.6 Mbps ATM Forum physical medium interface or an Ethernet interface. The ADSL LIM and the ANT contain the Discrete MultiTone (DMT) - ADSL transceiver and are sometimes referred to as ATU-C (ADSL Termination Unit-Central) and ATU-R (ADSL Termination Unit-Remote) respectively. This configuration is illustrated in Figure 2. Figure 3 shows the Access Network architecture for ADSL in more detail.

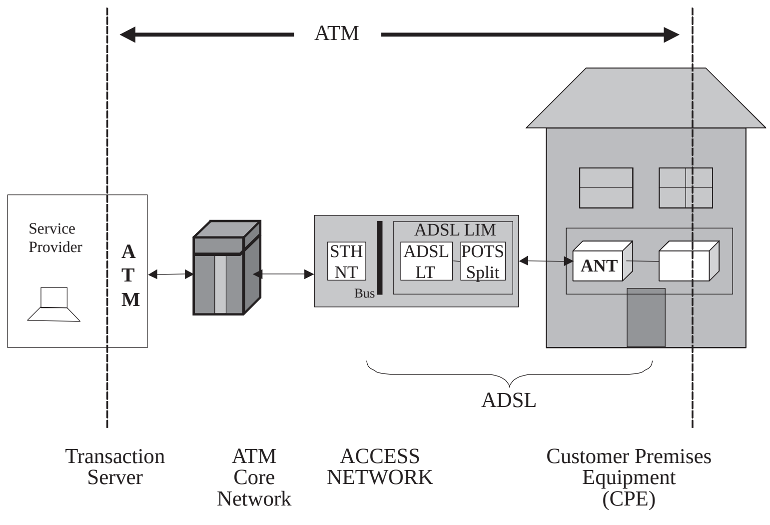

Figure 1. Bi-directional end-to-end ATM Connection

If only one SDH link is connected, the traffic is limited to 2.5 Mbps per ADSL line. This makes a total of 48x2.5 Mbps = 120 Mbps. This 2^{nd} generation LT can handle up to 5 Mbps downstream. Therefore, to be able to use 5 Mbps per line, two SDH STM-1 links have to be equipped or only 6 ADSL LIMs per access adapter can be inserted.

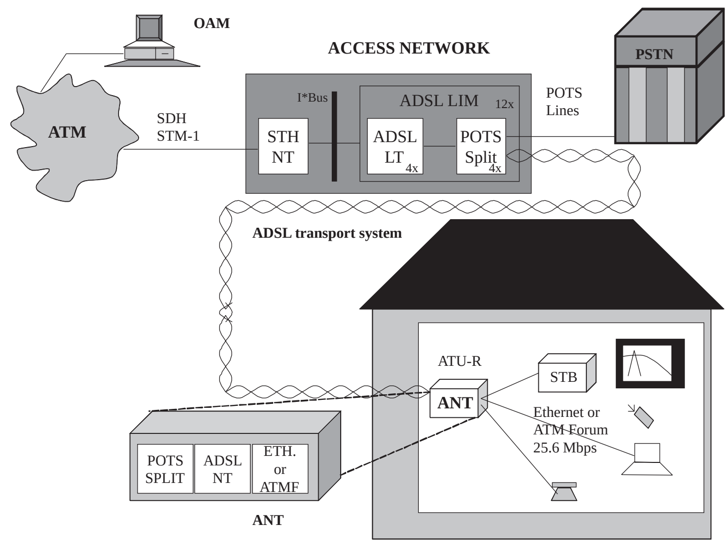

Figure 2. ADSL Configuration

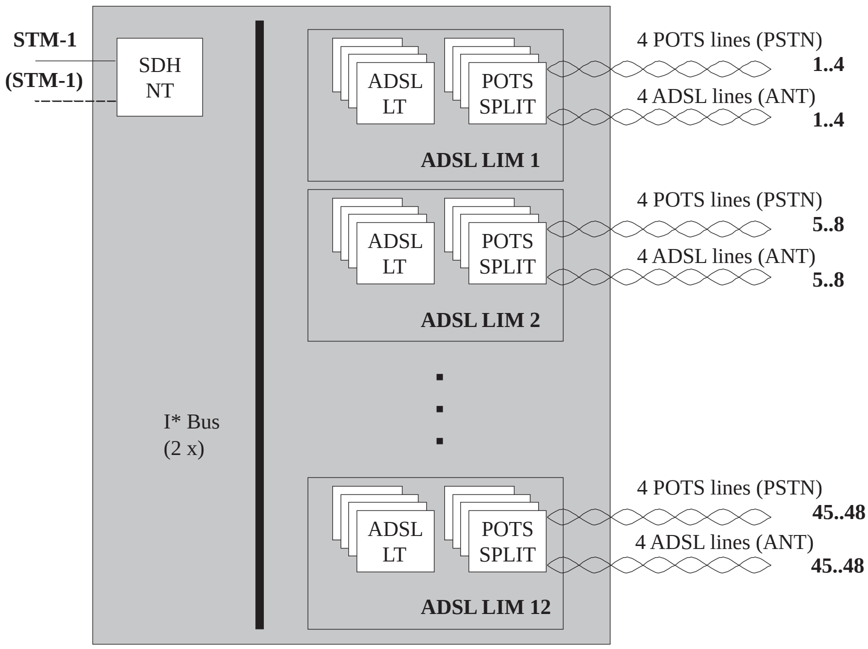

Figure 3. Access Network Architecture

3. Overview of SSCOP Flow Control

In ATM networks, we see that the ATM Adaptation Layer (AAL) is split into two subparts: a common part and a service specific part. The Common Part has two sublayers: AAL Segmentation and Reassembly (SAR) and Common Part Convergence Sublayer (CPCS). The Service Specific Part has two sublayers: Service Specific Coordination Function (SSCF) and Service Specific Connection Oriented Protocol (SSCOP). SSCOP is a protocol that has been approved as a Broadband The Integrated Services Digital Network (B-ISDN) ATM Adaptation Layer protocol standard, especially for use in the Signaling ATM Adaptation Layer (SAAL). The B-ISDN and ATM Forum standards require SAAL on the signaling Virtual Channel (VC);

bearer connections may or may not use any congestion mechanism, i.e. Request For Comments (RFC) 1577 specifies Internet Protocol (IP) / Logical Link Control (LLC) / Synchronous Network Architecture Protocol (SNAP) / ATM. SAAL incorporates many design principles for a high-speed protocol with fast processing operations. Integrated Services Digital Network (ISDN) Signaling structure can be seen in Figure 4.

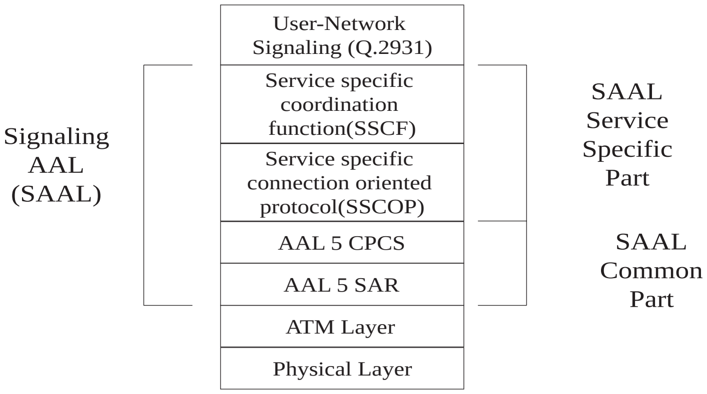

Figure 4. ISDN Signaling Structure

SSCOP, receiving Service Data Units (SDUs) from a signaling layer at different lengths, forms the PDUs and then transfers them to the peer SSCOP. At the other end, SSCOP conversely passes the SDUs to the signaling layer. SSCOP functions are sequence integrity, error correction by retransmission, flow control, keep alive, local data retrieval, connection control, transfer of user data, protocol control information (PCI) error detection, and status reporting [2].

In this paper, we will focus on the flow control aspect of the SSCOP. Flow control can be performed through the use of an adjustable sliding window. The receiver grants a credit window to the transmitter that allows it to transmit a certain number of frames. This credit value can be dynamically decreased or increased by the receiver.

In the SSCOP protocol, there are four types of frames: Sequenced Data (SD), POLL, STAT and USTAT frames. SD frames are used for transferring user data. The length of an SD frame is variable up to 65535 bytes. The transmitter sends a POLL frame periodically to request feedback from the receiver. In each POLL frame, there is a sequence number showing the sequence number of the next SD frame and a poll sequence number that is used as a timestamp. The receiver on the other hand replies with a STAT frame containing the highest (VR(MR)) sequence number that might be transmitted by the transmitter, the number of the next expected frame, the echoed poll sequence and a list of all currently outstanding SD frames. The transmitter updates its window according to the STAT frame it receives from the receiver. On the other hand, if a frame is lost, and the receiver understands that situation by checking the sequence numbers of the incoming frames, it immediately sends a USTAT frame, which requests the retransmission of the missing frame.

In AAL signaling, the POLL operation has three phases: Active, Idle and Transient. When a user is sending frames, the protocol produces POLL frames more frequently (the default is 750 ms). When the user does not send any frame for a while (the default is 2 seconds for example), this phase is called transient phase, and the protocol passes to the idle phase and sends POLL frames less frequently (default phase for idle phase is 15 seconds).

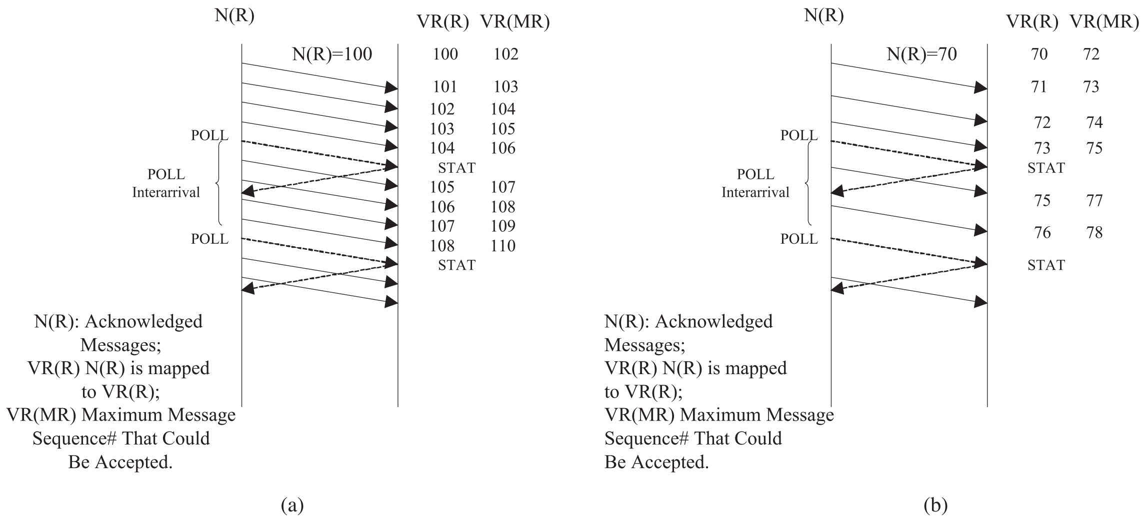

Figure 5. Example SSCOP Operation, (a) User Sending Packets Fast, (b) User Sending Packets Slow.

In Figure 5, an example SSCOP operation is illustrated for two different user types. In Figure 5-a, a user sends packets fast, so it reaches the VR(MR) value of 100 and is still sending packets at the same speed, while the other user in Figure 5-b sends packets more slowly and can only reach the VR(MR) of 70 in the same duration. However, as there are no precautions to prevent this situation, both situations are acceptable by the SSCOP protocol as long as the packets arrive at the destination without error.

As seen in the above example situation, the problem with this protocol is that it treats every call equivalently. That means it does not take into account calls coming with variable speeds. In other words, it does not have any precautions against users consuming the system resources unequally. Therefore, in the next sections, the buffer usage during normal and abnormal situations will be explained and the proposed method for controlling abnormal traffic will be given.

4. Window Size Needed For Saturated Transmitter

In [2], a detailed analysis is given for calculating the window size in an SSCOP protocol to keep the saturated transmitter from stalling. Based on these analyses, a window size can be chosen as follows:

In the absence of retransmission due to lost PDUs, a saturated transmitter will be kept active if the window size is

$$W = TR + TP$$

where TR is bandwidth-delay product and TP is the polling period, with both rounded up to *T=8(PDU_size)/PCR*. [2] The bandwidth-delay product is *Tr=roundTripDelay/T*. For the User Network Interfaces (UNIs) on the Access Node using the worst-case PCR of 1 Kbps from customer requirements and a PDU size of 300 bytes:

$$W = TR + TP = 2.4 + 2.4 = 4.8.$$

In the presence of retransmission due to a 10^{-7} bit error probability, the window needed to keep a saturated transmitter active is,

$$W = 2TR + TP + 1 = 4.8 + 2.4 + 1 = 8.2.$$

For the default UNI signaling channel Peak Cell Rate (PCR) of 500 Kbps, assuming the round trip delay due to ADSL is smaller than 0.1 seconds, W works out to be 2 and 2 for errorless and 10^{-7} error probability, respectively. Since it is not critical to keep the UNI SSCOP transmitter active, it is proposed to configure the UNI transmit windows to never be more than UniVrMrMax (default=2) (Maximum UNI VR(MR)). In addition the number of SVCs is limited to 16 per UNI, which reduces the need to keep the transmitter active.

The worst-case PCR for Automatic Number Identification (ANI) is also 1 Kbps, which results in $W = 8.2$ for 10^{-7} bit error probability. Since it is important to keep the Access Switch transmitter active, AniVrMrMax (Maximum ANI VR(MR)), the ANI window will be over-engineered. A default value of 16 is suggested as a starting point.

Allocating buffer space for outstanding credits results in about *2*692+16=1400* buffers. If "noisy" UNIs are isolated and the ANI signaling channel is well behaved, substantially less than 1400 buffers need be allocated, due to limiting factors discussed later. The maximum number of buffers is MaxSscopBufs, which is a common pool of buffers used by both ANI and UNI.

5. Buffer Usage During Normal Traffic

The number of calls in setup pending, MaxCallsSetup, is limited by the Signaling subsystem. If a maximum of 32 calls is allowed in the setup pending state, then subsequent incoming SETUPs receive an immediate response of RELEASE COMPLETE. Assuming a Poisson arrival rate for calls of 5 calls/second and an exponentially distributed network call setup time with mean equal to 5 seconds, the percentage of time spent in maximum setup pending mode can be calculated using Erlang's loss formula [3]:

$$p_m = \frac{\frac{\left(\frac{\lambda}{\mu}\right)^m}{m!}}{\sum\limits_{k=0}^{m} \frac{\left(\frac{\lambda}{\mu}\right)^k}{k!}}$$

where $m = 32$ and k is summed from 0 to m. The mean call arrival rate is specified by λ and the mean network setup rate by μ. Computing for 5 calls/second and a mean network setup time of 5 seconds indicates that the system will only be in setup overload 3 percent of the time. Recomputing for 20 calls/second indicates that the system will be in overload 68 percent of the time! Obviously, the number of calls in setup pending must be increased if users can generate traffic levels of 20 calls/second.

Each call in setup pending may consume at most 2 SSCOP buffers, which means that 97 percent of the time, less than 64 buffers are consumed by these calls. Assuming a Poisson arrival for RELEASEs and exponentially distributed network release times, a similar argument limits calls in release pending to 32 buffers. So in the absence of mass arrivals of RELEASEs at the ANI due to Base Station System (BSS) signaling restarts and other non-setup overload conditions, the SSCOP buffer pool is limited to 96 SSCOP buffers 97 percent of the time.

For a 3 percent overload time, an SSCOP buffer is held until the RELEASE COMPLETE is acknowledged. So the hold time on the buffer should be much smaller than the network setup time. Assume that maximum call setup lasts long enough to achieve equilibrium for arriving calls. Assuming that calls are still arriving at 5 per second while in maximum call setup, the acknowledge time on the RELEASE COMPLETE is exponentially distributed with a mean of one second, and there are at least m equal to 200 buffers available: Erlang's loss formula divides the denominator by a factorial (200), which results in underflow. Therefore, the probability that 5 calls per second arriving at busy Digital Subscriber Line Access Multiplexer (DSLAM) will exhaust 200 buffers is zero.

6. Proposed Throttling Mechanism

SSCOP specifies a dynamically sized transmit window. Credits for transmission are not automatically granted on acknowledgment of messages. Message acknowledgment and transmission credits are indicated in separate fields in the STAT and USTAT PDUs: N(R) (number of acknowledged messages) and N(MR) (number of transmitted messages). Advancement of the transmit window is indicated through the N(MR) field. Messages are acknowledged through the N(R) field. N(MR) is mapped to the VR(MR) state variable, which is the maximum message sequence number that the receiver can accept. N(R) is mapped to VR(R) state variable, which indicates that messages up to VR(R) are acknowledged at the receiver and buffers may be released by the transmitter.

Q.2110 states that the update of VR(MR) is implementation dependent [4]. The receiver can close the window by not incrementing VR(MR) or advance the window to any desired size at any time. A simple implementation is to increment VR(MR) each time SSCOP delivers a message to SSCF. One problem with this implementation is that a single user could exhaust the PDU buffer pool if the SSCOP runs in a task with higher priority than the Q.2931 task and the Q.2931 queue size is larger than SSCOP buffer pool [5]. Obviously, VR(MR) should not always advance during overload periods or for "noisy" users, so a more sophisticated implementation is needed.

The first level for protecting SSCOP buffers from destructive users is by stopping the flow of AAL5 PDUs or ATM cells. It can be shown that with exponentially distributed network setup times and the present constraint of 32 calls in the setup pending state, a single user sending 96 messages per second could keep Call Admission Control (CACO) in overload. Considering this and the fact that users are limited to 16 SVCs each, the limit of messages each UNI may generate must be limited.

6.1. At Network Interfaces

Mass setups are handled by MaxCallsSetup (Maximum Calls Setup). Mass releases at the ANI are quite possible due to signaling restarts in the network. The idea is to provide a system-wide variable, MinAniSscopBufs (Minimum ANI SSCOP Buffers), which signals the first low water mark for buffer usage from the common buffer pool. When this mark is reached, the ANI's VR(MR) is not updated anymore. When the SSCOP buffer pool rises above MinAniSscopBufs + AniVrMrMax, VR(MR) is once again incremented each time a message is successfully placed in Signaling's queue.

In summary, the strategy is to cease update of ANIs VR(MR) when MinAniSscopBufs is hit and increment VR(MR) for UNIs unless the interface is isolated.

6.2. At User Interfaces

For UNIs there will be one configurable limit defined: UniSscopHourlyPool (UNI SSCOP Hourly Pool) (default=200). The limit indicates the number of messages (SSCOP SDs) a user may send in a one hour period and not have their transmit window restricted. This would allow users to establish and release around 65 calls per hour before having their VR(MR) actively managed. UNIs remaining under the hourly limit have their VR(MR) incremented each time a message is placed upstream in the signaling queue.

Once a UNI exceeds the hourly limit, VR(MR) will be incremented by one every UniNumPollExpirys (UNI Number of POLL Expires) up to a maximum of UniVrMrMax. This method is shown in Figure 6, where UniSscopHourlyPool and UniVrMrMax are 100 and 2 respectively.

This proposed per-user throttling mechanism can be implemented efficiently. Each time an SSCOP connection enters the active signaling phase, the time-stamp for the last time the hourly credit pool was set to UniSscopHourlyPool is tested. If it has been more than an hour, the credit pool is initialized to UniSscopHourlyPool and a time-stamp is placed on the connection context. When the pool reaches zero, VR(MR) is updated every UniNumPollExpirys. This implementation causes no more interrupts by using the SSCOP events.

Users generating invalid sequences of messages will quickly be handled by the Signaling subsystem, which should limit the SSCOP buffer usage by such users.

The proposal is to place second watermark, other than the first one explained in the ANI part, MinUniSscopBufs, on UNI SSCOP buffer usage lower than the first watermark. The rationale is that malfunctioning Customer Promises Equipments (CPEs) are isolated and the ANI is more likely to legitimately exhaust SSCOP buffers. In a signaling restart scenario at the BSS, ANI cannot be allowed to exhaust the buffers otherwise RELEASE cannot be forwarded and RELEASE COMPLETE cannot be received at the UNIs.

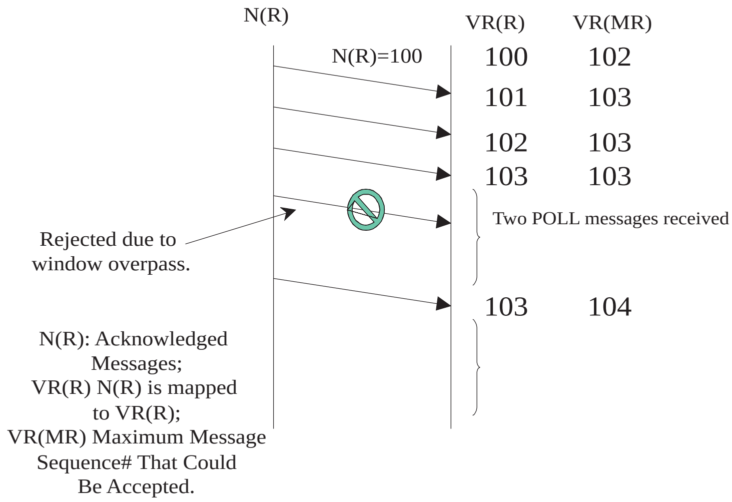

Figure 6. Proposed Isolation Mechanism

Hitting this UNI watermark has no effect except to raise an alarm (closing the UNI transmit window is counter-productive since RELEASE COMPLETEs could not be processed and calls and Common Transport Protocols (CTPs) will not be cleared from the system). If at any time SSCOP buffers are completely exhausted, then a defense action, such as shutting down SAAL on selected UNIs, may be appropriate.

7. Presentation of Simulation Results

The aim of the simulation is to see the effect of the proposed throttling mechanism on the performance of the system. For that purpose, the change in the delay and call rejection ratios have been calculated both with and without the throttling mechanism. The simulator is designed in an event-based manner.

7.1. System Description

The event list is designed as a First In First Out (FIFO) queue into which 577 lines are coming; 576 of these lines are the ADSL lines, and one is the feedback line. The SSCOP protocol, taking into account the service times of signaling and CACO, is the server of the system. The schema of the system is given in Figure 7.

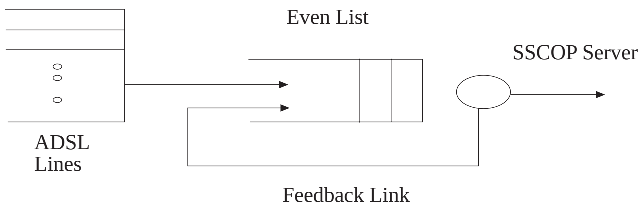

Figure 7. System Designed as a FIFO Queue

The packets coming to the system from ADSL lines are thought of as if they come from a Poisson Distribution. The program first initializes all lines with a SETUP event together with their occurrence times. These events enter the queue. The SSCOP Server takes the first event from the event list. If this event were a SETUP_REQUEST, either a PROCEEDING or a RELEASE COMPLETE would take place. That means if all 32 buffers are full or the same ADSL line has sent more than 16 calls, a RELEASE COMPLETE event occurs. Otherwise, another event (PROCEEDING) is formed and put into the event list. At that point, we need another exponential distribution to define the processing time of the SSCOP Server. For that purpose, we use a second exponential mean. In every situation, after a SETUP_REQUEST from an ADSL line is taken from the event list, another SETUP_REQUEST event is created from the same ADSL line. By doing so, we try to keep the ADSL lines feeding our system without interruption.

A buffer space occupied by a SETUP_REQUEST is released once a RELEASE event for that REQUEST comes to the server. At that time, we calculate the time that a call spent in the buffer. We use this information to calculate the usage of the buffer space by each ADSL line.

7.2. Adding SSCOP Functions

In Figure 8, we can see a flow diagram showing events.

In this diagram two more events are defined: RESTART and RESTART_ACK. However, these two events are neglected because they occur rarely. The numbers associated with each event show whether this event is an arrival to the Server or a departure from it. In this section, the time parameters between two consecutive events come from different exponential distributions. That means, in our first simulator, each interarrival time between events was coming from an exponential distribution representing the SSCOP Server. In other words, if the mean of the SSCOP Server was equal to 40 ms, the time between a SETUP and PROCEEDING packet was coming from the same distribution with the time between an ACKNOWLEDGE and RELEASE packet both having a mean equal to 40 ms. However, in this new version, the time intervals are adjusted according to the event triggering the next event. For example, the time between a PROCEEDING and a CONNECT packet comes from an exponential distribution with mean 1 second, while the time between an ACKNOWLEDGE and RELEASE comes from an exponential distribution with mean 1 hour. Therefore, a more realistic simulation platform is created.

- Means for time intervals between different events are as follows:
- From PROCEEDING to CONNECT: 1 s.

- From CONNECT to ACKNOWLEDGE: 66 ms.
- From ACKNOWLEDGE to RELEASE: 1.38 h, or 1 h 22 mins.
- Mean interarrival time for calls and SSCOP Server processing time are given to the system as variables.

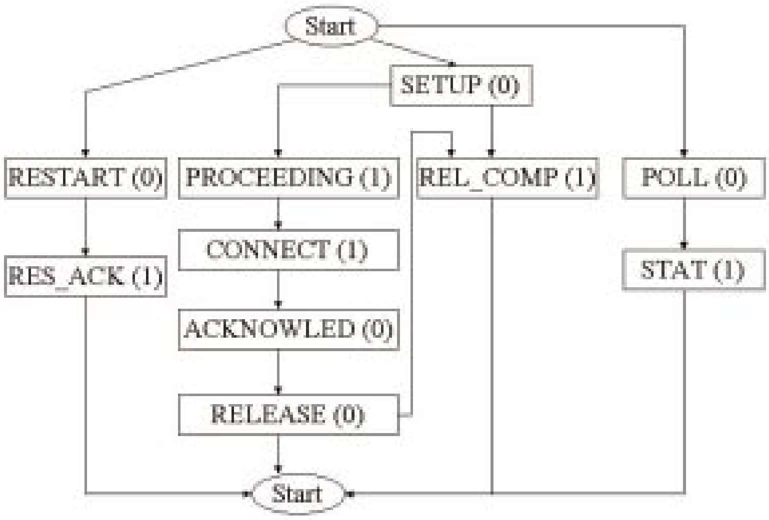

Figure 8. Event Flows.

POLL functions are created according to the data flow. For that purpose, the SSCOP connection is divided into two main phases and an intermediate phase. If the transmitter sends packets frequently, our Polling interval is 75 ms. That means if our Server receives two subsequent packets from the same user within 5 s, the system is taught to be in active phase, and the Server continues sending POLL messages each 75 ms. On the other hand, if the transmitter does not send a packet for 5 s, which is accepted as a transient phase, then the receiver starts a passive phase, and sends a POLL message every 15 s. Our SSCOP Server, once it receives a POLL message, creates both another POLL message to be put into our future event list, and a STAT packet. These two packets are sent back to our future event list or, in other words, into our FIFO queue. The system starts with the passive phase.

The test is performed to test a time period of 24 hours. During the test, the values that we are looking for are as follows:

1. Total delay (T) at each ADSL Line,
2. Total time of call duration at each ADSL Line staying in the system,
3. Number of accepted calls,
4. Number of packets rejected,
5. The number of calls rejected because of our limited buffer space,
6. Number of packets created,
7. Number of calls rejected because the user has already created 16 SVCs,
8. Number of POLL packets created at each ADSL Line.

7.3. Adding Malfunctioning Users

The last step of the simulation is adding some malfunctioning users so that we can see the effect of the new isolation mechanism. For that purpose two different types of malfunctioning users are created. The first type of malfunctioning user sends packets faster than the other normal users. This type of user, called an irregular user, illustrates the users who try to send more packets than their contract. In other words, they pay for less bandwidth but use more, and in that way consume the resources belonging to some other user.

The second type of malfunctioning user sends irregular calls, which means in a normal call the packet order is given in either {SETUP, PROCEEDING, CONNECT, ACKNOWLEDGE, RELEASE, and RELEASE_COMPLETE}, or {SETUP, RELEASE_COMPLETE} way. POLL and STAT packets are not taken in account into a normal call definition. However, a malfunctioning user, instead of sending packets in the described manner, sends a SETUP packet followed by a RELEASE packet. In that situation, a SETUP packet is accepted into the buffer. While it is being processed, a RELEASE request comes to the server. Therefore, the server waiting for a PROCEEDING packet receives a RELEASE packet. This order is called irregular. As a result, an irregular user sends packets in an irregular fashion. The aim of such a user would be either testing the system in worst cases, or crashing the system. The event flow for a user sending packets in irregular fashion is given in Figure 9.

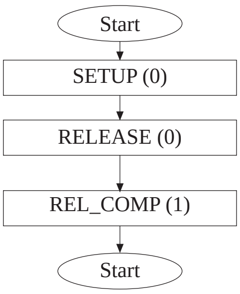

Figure 9. Event Flows for an Irregular User

Another property of malfunctioning users is sending packets in either a normal sequence or as only a SETUP-RELEASE sequence according to a fixed probability. For that purpose, the malfunctioning user picks a random number before making a SETUP request. If this number is lower than the predefined probability, then it sends a SETUP packet followed by an immediate RELEASE packet. Otherwise, it makes a normal connection. In the following examples, we used only this kind of malfunctioning user definition in some tests.

7.4. Test Results

In the first test, we compared the delays of an expected call connection and the delay occurring because of the isolation mechanism (IM). This result is shown in Figure 10.

In this figure, the x-axis shows the ADSL Lines, and the y-axis is the delays in seconds. As seen, the delays caused by the IM are very low compared with the connection delay. That means it does not add too many burdens on the protocol.

In the second test, the rejection ratios of calls are calculated. As seen in Figure 11, nearly 85% of calls are being rejected because of the flow control mechanism. In the diagram the rejection ratios of two flow control mechanisms are also shown. Approximately 20% of calls are rejected by the IM mechanism while the rejection ratio of calls in a system with only 16 SVC limit flow control is nearly 40% . That means if we use both mechanisms, we reject 60% of calls. In that situation, an improvement of 20% can be obtained by the new IM mechanism. In this figure, the call rejection ratio at ADSL line 8 is much lower than the rates at other lines. The reason is that this user sends packets in an irregular way. Therefore, they donot stay too long on the system. As a consequence, even if the traffic control mechanism slows down the user at that line, it still sends much more packets than the other, and leaves the system nearly immediately. Therefore, it is nearly impossible to create more than 16 SVCs at the same time. That means the 16 SVC flow control mechanism oes not take effect in that situation. The only traffic control scheme that would slow down that user is the IM mechanism. As the rejection ratios because of IM are very small, the total rejection ratios are also very small in that situation.

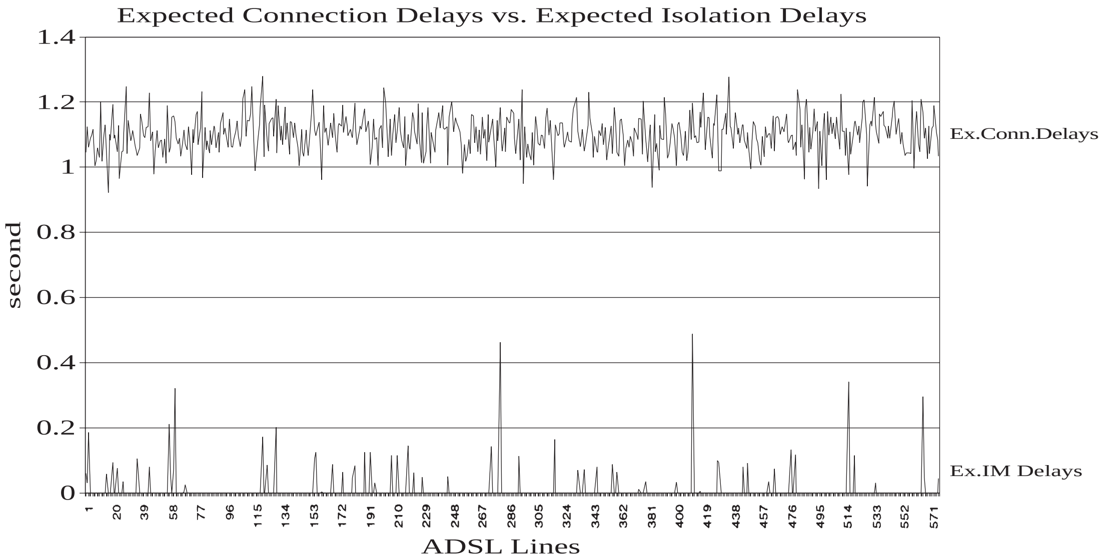

Figure 10. Comparison of Expected Call Connection Delays with Expected Packet Rejection Delay because of IM Procedure

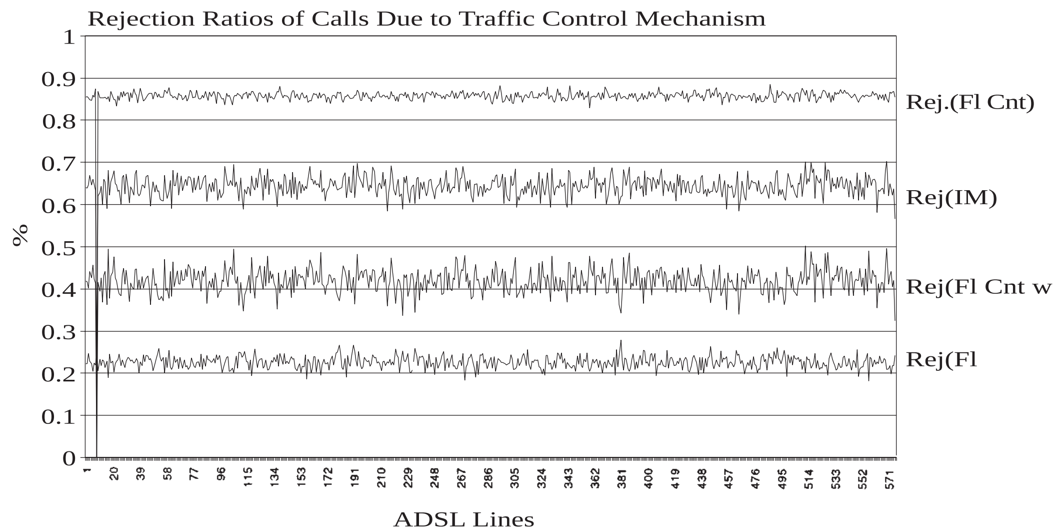

Figure 11. Comparison of Call Rejection Ratios for Different Flow Control Techniques.

In Figure 12. Call Acceptance probabilities for the two flow control mechanisms are compared. As seen in this figure, a 20% improvement can be obtained with this method. The only exception occurs in line 8, at which a second type malfunctioning user exists. A similar effect with the previous test can be seen in that test also with user 8. That means the user at ADSL Line 8 sends packets in an irregular way; therefore, that user, sending faster and in an irregular fashion, can not be rejected by 16 SVC limitation. As a consequence, the traffic control mechanism accepts most of the packets coming from that line.

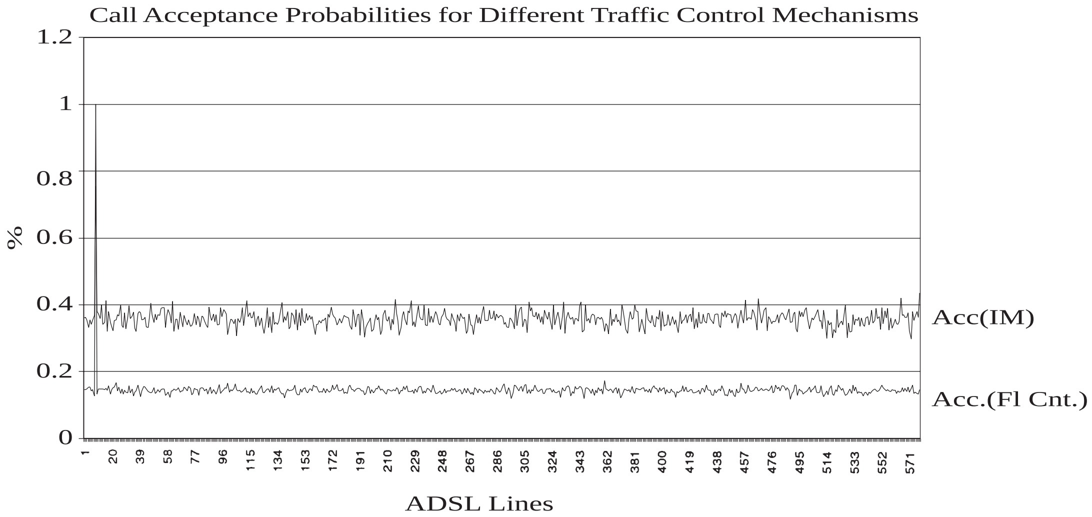

Figure 12. Comparison of Call Acceptance Probabilities

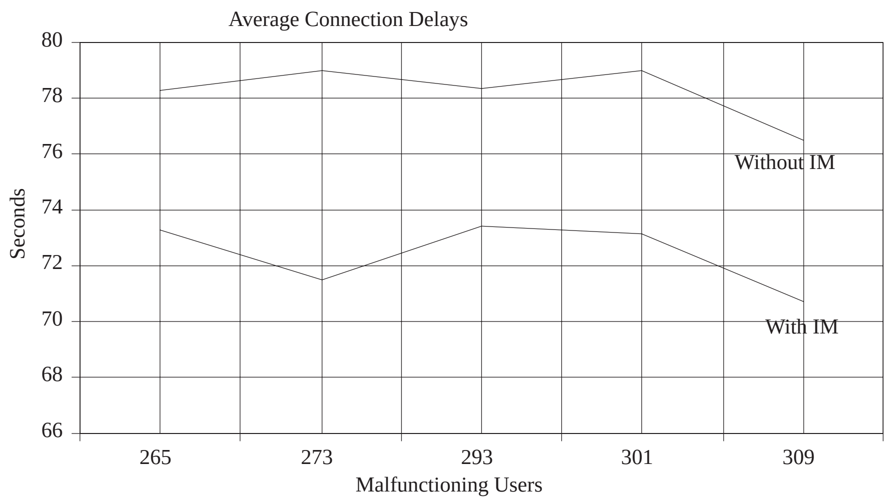

Figure 13. Average Connection Delays with random number of malfunctioning users.

In the next test, the number of malfunctioning users is chosen randomly. For each one, average connection delays and average traffic control delays are calculated. These two results are shown in Figures 13 and 14 respectively. As seen in these figures, the connection delay is approximately 73 seconds with IM and 78 seconds without IM while the delay for traffic control is nearly 1 second with IM and 2 seconds without IM. That means the delay for traffic control is short compared with the connection delay. In other words, traffic control does not put an extra burden on the protocol, even if the number of malfunctioning users increases. The delay spent for traffic control nearly doubles when the IM mechanism is not used. At the same time, the connection delay could also be decreased approximately 7% with IM.

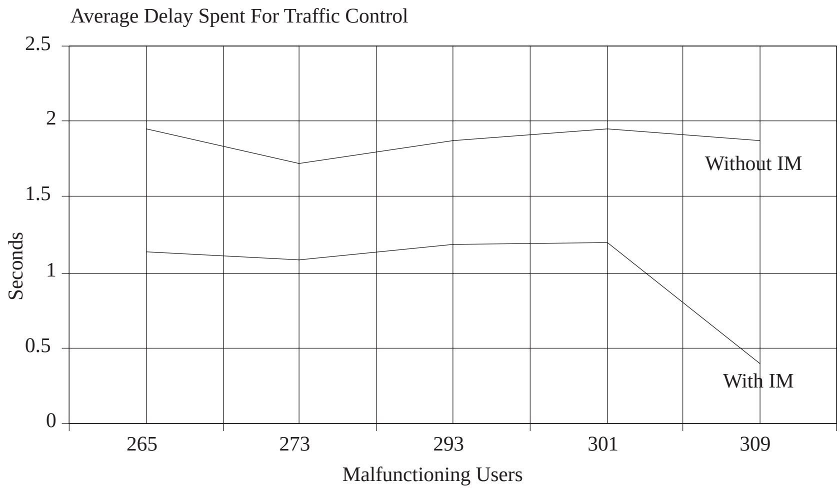

Figure 14. Average Delays Spent for Traffic Control with random number of malfunctioning users.

The last figure, Figure 15, shows the average call duration for the same system. It is seen in that diagram that the average call duration decreases from 7.56 to 4.91, which means a decrease of 35% when the IM mechanism is used. This last scenario demonstrates the effectiveness of the IM, because it shows the total performance increase, not only on one step. That means when we only look at the connection delays, we donot take into account the delays occurring on another step during a normal communication such as the delay between a CONNECT and an ACKNOWLEDGE, the delay between a RELEASE and a RELEASE COMPLETE, and so on. This last diagram, on the other hand, gives us a general picture, the change in the whole connection time when IM is used.

8. Conclusion

In this paper, a new congestion control mechanism is proposed for the SSCOP protocol in signaling. It is explained that this mechanism, outperforms the existing congestion control mechanism which is limited only by the number of calls that would be created simultaneously. The system is designed to work on ADSL connections signaling over the UNI. The next step would be to find out the total system performance metrics adding the CACO and SAAL level processes and also trying to analyze the system analytically.

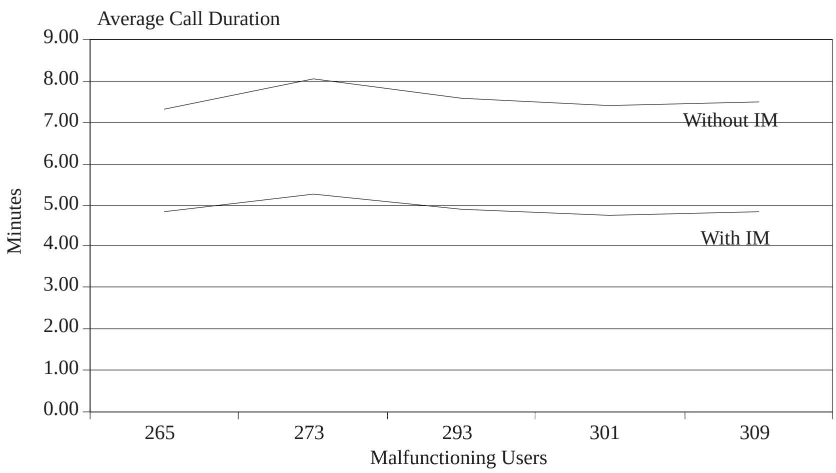

Figure 15. Average Call Duration for Traffic Control with random number of malfunctioning users.

References

[1] Onvural, O.R., *Asynchronous Transfer Mode Networks, Performance Issues*, Second Edition, Arctech House, London, 1995.

[2] Henderson, Design Principles and Performance Analysis of SSCOP, *COMSAT Laboratories*, 1991.

[3] Kleinrock, *Queuing Systems Volume I: Theory*, Wiley, 1975.

[4] ITU-T Recommendation Q.2110.

[5] ITU-T Recommendation Q.2130.