Flow-control Issues in ATM Signalling Link Deployment

K. Kant
Bellcore, 331 Newman Springs Road, Red Bank, NJ 07701, USA

Abstract

This paper examines certain performance issues concerned with the overallocation of bandwidth for ATM signalling links in a common channel signaling (CCS) network. In particular, in several deployment scenarios, a signalling link may be allocated more bandwidth than the capacity of the processors would allow, thereby causing congestion. This study indicates that a link-level flow-control alone is inadequate for an acceptable overload performance and must be augmented with appropriate congestion controls at each protocol layer. Currently, CCS network elements typically do not provide an effective congestion control at MTP3 and SCCP layers. This study indicates that such controls must be in place in order to protect network integrity. The paper also provides some guidelines for setting these controls, and highlights potential dangers of depending too much on them.

1 Introduction

This paper examines the impact of high capacity ATM based signalling links in a narrowband or broadband common channel signalling (CCS) network. In particular, we are concerned with signalling links based on the Signalling ATM Adaptation layer (SAAL) suite of protocols, and consider situations where the bandwidth allocated to a SAAL link is significantly higher than the available processing resources to handle this bandwidth. (Clearly, the link-level processors should be able to handle a fully loaded signalling link; it is the capacity of the higher level processors that is of concern here.) Thus, during periods of overload, the SAAL link may bring in a considerable amount of traffic and overload these higher level processors. On the first thought, it appears that link-level flow-control can be used to protect the higher level processors in such situations. This paper shows that flow-control alone is inadequate and must be complemented with processor level congestion control to obtain acceptable overload performance. The application layer in this paper is assumed to be either ISUP or BISUP which attempts to set up a point to point "call" or "connection" for communication. For generality, the paper assumes quasi-associated signalling, although the results apply equally well to associated mode signalling. It may be noted here that broadband networks are initially expected to be deployed with associated mode signalling.

There are several reasons why the higher level processors may be unable to handle a fully loaded signalling link. One major reason is that the signalling link speed determines a number of engineering parameters such link congestion thresholds, error monitor parameters, link proving parameters, etc., and it is difficult to support a large number of link speeds from an operations and management perspective. Thus, even though the ATM technology allows the link speed (i.e., the virtual circuit bandwidth) to be chosen arbitrarily, only a few speeds are likely to be used in practice. In fact, in the foreseeable future, SAAL links are likely to use only T1 speed (or E1 speed in Europe). (Broadband networks are unlikely to use 56/64 Kb/sec links because even a 1.5 Mb/sec signalling link accounts for only 1% of a 150 Mb/sec ATM pipe.) Consequently, the link bandwidth could well be much higher than the required signalling load. Since it is more reasonable to consider the latter for choosing processor speeds, the processors may be unable to handle fully loaded signalling links during periods of overload. In the context of current CCS network, the deployment of a SAAL link may well be dictated by the lack of ports, rather than by a quantum jump in the traffic. For example, replacing a linkset containing 3 56 Kb/sec links by a single 1.5 Mb/sec SAAL link frees up 2 ports. In this scenario, it is unreasonable to upgrade higher level processors unless the freed up ports are expected to carry a lot of new traffic.

The higher level processors alluded to here could include MTP3 (or level 3) processor (responsible for routing, distribution, network management) and processors for "MTP3 user" layer (e.g., SCCP, ISUP/BISUP, TCAP, etc.). Typically, mechanism such as ACC (automatic congestion control) and ACG (automatic code gapping) are usually in place to handle (B)ISUP/TCAP congestion. Handling of MTP3 and SCCP congestion, although defined, are not usually implemented in current switches. In this paper, we specifically focus on the consequences of not doing an effective MTP3 congestion control, although similar results should apply in the SCCP context as well.

The SAAL protocol provides a flow-control scheme to ensure that the receive-end of the link is not overwhelmed. This scheme can be used to control the amount of incoming traffic, and thereby prevent overloading of level 3 processor. The study in this paper shows that an explicit congestion control at level 3 is still essential for an acceptable performance. However, level-3 congestion control may be undesirable because it results in a global throttling of the traffic. The study here indicates that by using both level-3 congestion control and flow-control simultaneously, one can avoid triggering level-3 congestion control often, while obviating the need for making the flow-control very precise.

The outline of the paper is as follows. Sections 2 and 3 discuss SAAL protocol (including flow control) and MTP3 congestion control respectively. Because of complex interactions involved in flow and congestion controls, the paper studies the problem entirely by means of a simulation model. The simulation model and study results are described in section 4. The paper assumes that the reader is familiar with SS7 architecture, congestion control and related issues. Reference [6] and articles contained in [7] provide some of the background material on the subject.

2 SAAL Protocol and Flow-Control Schemes

The SAAL protocol is a selective-receive type of ARQ (automatic repeat request) protocol with periodic polling of the receiver by the transmitter. In response to a *poll* message, the receiver sends a *stat* message, which, among other things, contains the highest sequence number up to which the transmitter can transmit. That is, the receiver periodically, provides a *credit* to the transmitter. The precise details of credit control are considered to be implementation dependent and are not specified in the protocol. Reference [5] specifies two credit-control schemes and studies their parameterization. Of these, the *fixed credit* scheme appears to provide a somewhat better control is used for the purposes of this paper. In this scheme, credit is controlled explicitly (i.e., even if the receive buffer is not full) over an allocation period of b_f polling intervals. The precise parameter settings are as follows:

1. Credit allocation block size (b_f) of 3 polling intervals. (The polling interval T_{poll} itself is chosen as 0.1 seconds.)

2. Jitter in successive stat arrivals (δ_s) of 0.05 seconds, and end-to-end transit delay of polls and stats (τ_u) of 0.035 seconds.

Let C_{l3} denote the credit required according to level 3 processor considerations. For this, we start with C_{bl3}, defined as the number of PDUs arriving during a credit allocation interval. That is,

$$C_{bl3} = r_{engr}(b_f T_{poll} + 2\tau_u) \tag{1}$$

where r_{engr} is the PDU rate on the SAAL link under normal conditions (i.e., when level 3 processor operates at the nominal engineered utilization level), and the factor $2\tau_u$ accounts for the time lag between the SSCOP status between the transmit and receive ends. We then compute C_{l3} by scaling C_{bl3} upwards so that level 3 processor utilization is at the maximum allowable level. Thus, if the SAAL link is given a credit of C_{l3} every credit allocation interval, the level 3 processor will not be overwhelmed. (It is assumed here that link capacity exceeds that of processor capacity. In general, the credit granted would be the minimum from link and processor considerations.)

The above credit control scheme is *static* since the credit amount is treated as a provisionable parameter that does not depend on instantaneous loading of level 3 processor. The obvious alternative is to do an adaptive credit control based on processor utilization, queue length, or other performance measures. Unfortunately, adaptive control is no panacea because of the need for status communication from level 3 to level 2, and the fact that a level 3 processor gets traffic from local level 4 and other level 3's and such traffic is not controlled. Thus, an adaptive control may be both complex and result in unfairness and perhaps oscillations. In view of this, we shall concentrate on static control only.

Both static and adaptive credit control must deal with the lag between transmitter and receiver (in terms of both propagation and queuing delays), as can be seen from the equations above. Consequently, the credit granted may sometimes be too little (which, as we shall see, is always undesirable) or too much (thereby overloading the level 3 processor). That is, a precise credit control is difficult.

3 Level-3 Congestion Control

There are two general ways of limiting the incoming traffic to the level 3 processor:

1. Regulate the scanning mechanism used for picking up PDUs from various level 2's served by this level-3 processor.

2. Use MTP3 congestion thresholding so that the messages exceeding the threshold are dropped and a TFC message is sent to the MTP3 traffic source.

Current switches implement neither of these controls. The first scheme can be viewed as a form of adaptive credit control discussed earlier and thus suffers from the same problems, i.e., having to deal with uncontrollable traffic coming from level 4 and other level 3's. Therefore, we shall concentrate on MTP3 congestion control scheme only. More specifically, we consider the ANSI version, which provides separate congestion thresholds for each congestion priority level. This scheme is similar to link congestion control except that a TFC is not sent on every message that must be dropped, but instead on every Nth dropped message. In this study, we choose $N = 8$, which is the recommended default value. For simplicity, we also put the onset and discard levels at the same point.

The major difficulty in implementing MTP3 congestion control is to determine appropriate congestion thresholds because they depend heavily on the architectural details of level 3. The task becomes easier if we think of congestion thresholds in the units of seconds, rather than messages or bytes. For example, if level 1 onset threshold is set at 0.3 seconds, it means that we are willing to accept a maximum message delay of 0.3 seconds, before we take action to cut down the traffic and bring the delays down. Most of the fundamental considerations in setting thresholds concern time units, such as delay tolerance of the applications, feedback delays (which determine how quickly the traffic can be controlled, and hence how much uncontrolled traffic will be received), potential for starvation (which has to do with how long the traffic is throttled), etc. These time-thresholds could be converted to messages/bytes by vendors or network operators depending on the message service rates, message sizes and available buffer space. A detailed derivation of congestion thresholds is beyond the scope of this paper; here, we choose

Figure 1: Network Model used in Simulation

these thresholds based on the corresponding link congestion thresholds given in Bellcore GR2878 [2]. The chosen abatement and onset thresholds (in seconds) for various congestion levels (denoted A_i's and O_i's) and the total buffer size S are as follows:

A1	O1	A2	O2	A3	O3	S
0.26	0.62	0.99	1.28	1.58	1.73	1.88

These thresholds perhaps lead to buffer size that is a bit too large, however, but we shall not pursue this aspect. The qualitative conclusions in this paper do not depend on the precise threshold values.

4 Simulation Model and Overload Experiments

Fig. 1 shows the (quasi-associated) signalling network model used in the simulation. It consists of 3 subnetworks, each with 6 SSPs and one STP. The simulation model implements, in detail, SS7 message flows for point-to-point ISUP/BISUP calls, MTP3 and SAAL protocols, flow-control, congestion control, calls via intermediate switches (important from a congestion-control perspective), customer behavior, and an accurate accounting of the impact of limited trunks and subscriber lines on the signalling performance. The model engineers the network automatically during initialization phase based on target utilizations and other engineering parameters.

For the purposes of this paper, the network shown in Fig. 1 is engineered such that under normal conditions every processor and link (except STP level 3 processors) operates at a utilization between 0.2 and 0.25. The level 3 processors operate at a much higher utilization if they are intended to be underengineered, otherwise, they operate at a utilization of 0.2. The main motivation for using a rather low nominal utilization is to ensure that there are no unexpected bottlenecks during overload periods. Also, during normal operation, the calling patterns are completely symmetric so that no "hot-spots" are created.

Since the link/MTP3 congestion indications do not go past the first MTP3 user layer (such as BISUP or SCCP), traffic throttling may not happen at the source if the calls go via intermediate switches. This leads to a poorer overload performance. In order to account for this, we use a mix of calls in the simulation that go via 0, 1 or 2 tandems. Also, in order to keep the analysis valid for current and future traffic mixes, we use both simple and SCP-assisted (including AIN) calls.

A particularly important aspect in this study is the impact of customer behavior and voice network on the performance. We model customer behavior according to measurements reported in [3] except for a somewhat larger post-dial abandonment delay. The number of trunks in the network are engineered so that trunk congestion happens only rarely. The number of subscriber lines at a switch are chosen as the average number of busy lines when the call processor is running at a utilization of 0.4. During the simulation, every number is equally likely to be called (i.e., every subscriber line becomes equally likely to be requested by an incoming call). This ensures that the probability of callee-busy increases with the number of calls in progress.

It is appropriate at this point to make a few comments regarding the architecture used in the simulation. Our model has a centralized level 3, which means that all links in the STP will be served by a single uniprocessor or multiprocessor system. Most of the current STPs use a more distributed architecture, where the level 3 consists of a number of cooperating subsystems, each of which handles only a small number of links. In certain architectures, there is a dedicated level 3 processor for each link. We believe that our results apply to all these architectures, at least qualitatively. For example, suppose that in Fig. 1, each link has a dedicated level 3 processor. In such a case, the flow-control over the SAAL link will impact only inter-subnetwork calls. (Actually, our model doesn't have any intra-subnetwork calls.) Since every message for an inter-network call must go through the level 3 processor of SAAL link, the impact of flow-control and resulting transmit congestion on the other end of the SAAL link will still be experienced by every such message.

The simulation starts with normal operation and is run in the normal mode for the first 30 seconds of simulated time, at which time it goes into overload. The overload period lasts 3 minutes. The simulation model reports statistics collected over successive 15 second blocks of time so that it is possible to track the behavior as a function of time. It is found that the performance typically continues to degrade and settles down only in the last few blocks. However, for the purposes of comparison, the overall performance over the entire overload duration may be of more interest, since most overload periods may last only for a few minutes.

During the overload period, the calling rate of all SSPs is increased by a factor of 1.5. In addition, the call routing becomes highly asymmetric in order to create a focussed overload towards STP1. In particular, 80% of calls originating at subnetwork 2 go towards subnetwork 1, and the remaining 20% towards subnetwork 3. Similarly, 80% of calls originating at subnetwork 3 go towards subnetwork 1, and the remaining 20% towards subnetwork 2. This is rather severe overload situation and should cover most overload scenarios arising in practice. Note that a non-focussed overload scenarios would not be as interesting because they would tend to create bottlenecks in other processors and other network elements as well.

With the above setup, the node of primary interest is STP1, and the calls of primary interest are those that originate or terminate in subnetwork 1. In this context, the following two cases are of interest:

1. Level 3 processors in STP2 and STP3 have adequate capacity to handle a fully loaded SAAL link (perhaps because those STPs are supplied by a different vendor or because subnetworks 2 and 3 belong to a different service provider).

2. All three STPs have an underengineered level 3. (More specifically, we assume that all 3 STPs are identical in our simulations).

The next two subsections analyze these two cases in detail. For this, we ran experiments by using credit amounts of ηC_{l3}, with the multiplicative factor η in the range of 0.5 to 1.5. The η value that yields the optimal performance will henceforth be denoted as η^*. It will be seen that η^* depends

Figure 2: Call setup performance vs. flow-control parameter η

on how underengineered the level 3 processor is. Henceforth, we denote the underengineering amount by the symbol α. Unless stated otherwise, we choose $\alpha = 0.5$, which means that the capacity of level 3 processor is only 50% of what is necessary in a well-engineered scenario.

4.1 Level 3 Capacity Limitation Only at STP1

We studied the effectiveness of flow control in this situation by restricting flow in link directions 21 and 31 by choosing various values of η. No significant credit restriction was enforced in the link directions 12 and 13.

Before discussing specific results, let us discuss the trend to be expected as a function of η. If $\eta < \eta^*$, the traffic is being restricted unnecessarily, which will result in a poor throughput, large setup delays (due to transmit congestion on the other end), and a large number of failed calls. As η approaches η^*, the performance will improve significantly, however, the level 3 processor queue will start to build up. For $\eta > \eta^*$, the performance will suffer again due to level 3 processor congestion. Curve 1 in Figures 2(a) and 2(b) shows this behavior for η ranging from 0.5 to 1.5 (for $\alpha = 0.5$). It turns out that at $\eta = 1.2$, there is almost no flow-control (no-credit period reduced to 6.3 seconds over the 180 second overload period). It is clear that setting η close to 0.9 yields good throughput and low call setup delays; however, the behavior is not very stable: a small error in credit estimation or a small change in network parameters could lead to overloading of the level 3 processor and very poor call setup/release delays. Similarly, if η is chosen close to 0.6, small changes in network parameters could again result in poor call setup delays. (The call throughput is already pretty bad for $\eta = 0.6$.) Thus $\eta \approx 0.75$ is the best choice in this case.

Curve 3 also shows performance without MTP3 congestion control but with $\alpha = 0.33$, i.e., when level 3 processor has only 1/3rd of the desired capacity. It can be seen that η^* is still about 0.9 and the behavior is qualitatively similar to that for $\alpha = 0.5$ except for much higher call setup delays. However, for $\eta < 0.8$, a very unexpected behavior results: the size of level 3 processor queue increases with decreasing η. In other words, a tight flow-control serves to overload the level 3 processor instead of protecting it. The reason for this anomaly is as follows: A very tight flow-control results in severe transmit congestion for link directions 21 and 31. This results in dropping of high priority messages like ANM and RLC for calls originating at subnetwork 1. Consequently, a large number of calls fail and are retried. The reattempt traffic places even more load on level 3 processor, thereby further aggravating the situation. In fact, the high reattempt traffic even results in trunk congestion at the tandems of subnetworks 2 and 3. Because of this effect, the range of η over which call setup delay is acceptable becomes quite small for $\alpha = 0.33$. Because of difficulties in precise credit control, this means that no value of η is likely to provide acceptable performance in practice.

Curve 2 and 4 in Figures 2(a) and 2(b) show the performance if MTP3 congestion control is implemented. As expected, MTP3 congestion control stabilizes performance for large η. In fact, with $\eta \geq 1.2$, when there is almost no flow control, one gets pretty good call setup delays and call throughput. The only advantage offered by having flow-control is a significant gain in throughput (and corresponding decrease in failed calls) at the expense of somewhat higher call setup/release delays.

The main conclusions from these results for Case (1) (i.e., when only STP1 has limited level 3 processor capacity) are as follows:

1. To achieve optimum performance without MTP3 congestion control, credit determination must be very precise. Unfortunately, the latter is very difficult because of uncertainties in several quantities such as the amount by which transmitter runs ahead of the receiver, current link length, queuing delays suffered by stats/polls, etc.

2. With MTP3 congestion control in place, flow-control is not really required since a suitable choice of congestion thresholds can provide both a decent throughput and low setup delays.

3. When the level 3 processor is severely underengineered, a very tight flow-control can backfire and may *cause* the level 3 processor to be overloaded. This observation again calls for a precise credit control in the absence of MTP3 congestion control.

4. The global control exercised by the TFC mechanism and the difficulty in properly setting MTP3 congestion thresholds makes the dependence on MTP3 congestion control undesirable. In contrast, the effect of flow-control is very localized. Thus, it appears that the best mechanism might be to use MTP3 congestion control along with a somewhat loose flow-control.

4.2 Level 3 Capacity Limitation at all STPs

In this case, we under-engineer level 3 processors of all three STPs by the same factor α, and restrict credit in both directions of STP-STP links. For small η (< 0.6 for $\alpha = 0.5$ and < 0.7 for $\alpha = 0.33$), the experiments show a very peculiar behavior: as the system goes into overload, the number of successful calls from subnetwork 1 goes down quickly. After about 90 seconds of overload, subnetwork 1 experiences a “target silencing” phenomenon such that incoming calls to subnetwork 1 are processed, but outgoing calls are not allowed at all.

This phenomenon can be explained as follows: The transmit buffer of a SAAL link receives two types of call setup traffic: (a) IAMs corresponding to call attempts generated by local SSPs, and (b) ACM/ANM transmitted in response to IAMs coming from the subnetwork on the other end. As the overload period is entered, both links 12 and 21 get lot more traffic, but the nature of traffic is different. Because of the focussed overload towards subnetwork1, link 21 gets a lot of type (a) traffic, whereas link 12 gets a lot of type (b) traffic. Since type (b) traffic is generated in response to type (a) traffic on the other end (and hence has some lag), the transmit congestion builds up on link 21 faster. Eventually, when level 1 transmit threshold is crossed, STP2 generates TFCs which restricts type (a) traffic at STP2. A similar thing happens at STP1; however, since STP1 has a lot more type (b) traffic, the transmit congestion at link 12 continues to rise. This explains the higher congestion level (i.e., 2) for link 12 compared to that for link 21 (i.e., 1). The effect of this higher congestion level is that almost all IAMs generated by subnetwork1 get dropped (since they have congestion priority 0), whereas the ACM/ANM don't suffer (since they have congestion priorities > 0). The result is an eventual shutdown of all calls generated by subnetwork1.

Given that $\eta \leq 0.7$ to avoid MTP3 congestion, and $\eta > 0.7$ to avoid target silencing, we are left with no choice but to use MTP3 congestion control. Figures 2(c) and 2(d) show call throughput and setup delay performance for Case 2. It is seen that without MTP3 congestion control (curve 1), the call setup delays remain high for all values of η. With MTP3 congestion control (curves 2 and 3), a lack of flow-control gives a reasonable performance. However, a loose flow-control can provide a better throughput at the cost of somewhat higher (though still acceptable) call setup delays. The results show that looser flow-control becomes more desirable as α decreases. It is also found that the calls originating from subnetworks 2 and 3 experience performance similar to those originating from subnetwork 1 for $\eta > 0.9$. These results can be summarized as follows:

1. MTP3 congestion control is essential if the credit must be restricted in both directions of a SAAL link.

2. If the level 3 capacity limitation is not too severe, a loose flow-control (as given by equations in section 2 with $\eta = 1.0$) gives the best perfor-

mance. However, with a very severe capacity limitations, it is better not to impose any flow-control at all.

5 Conclusions

This paper presents a simulation study which examines the problem of ensuring network integrity when the SAAL links in a signalling network provide much higher throughput than the ability of higher level processors to handle it.

The study indicates that an attempt to protect level 3 processor by means of flow-control alone is unlikely to be successful, and MTP3 congestion control is required for robust and acceptable performance. Although, MTP3 congestion control alone yields acceptable performance, the global throttling provided by it is undesirable. Therefore, a combination of loose flow-control coupled with MTP3 congestion control appears to be the best choice. Good MTP3 congestion control is difficult and depends heavily on the architectural details. In this context, the paper provides some simple guidelines for setting MTP3 congestion thresholds, but much further work is needed on this topic. The simulation model in this study considered a centralized level 3. It is important carry out studies for other level 3 architectures as well. Finally, congestion controls at the SCCP level should also be examined, particularly in view of increasing SCP loads due to complex AIN services.

Further work on the subject includes establishing good SMH congestion controls and studying the performance of adaptive credit control schemes. Adaptive credit control schemes may be useful if they can provide a significantly better performance with only a minimal amount of reporting and coordination between levels 2 and 3. It is also essential to perform the experiments in this paper for non-centralized level 3 architectures.

Acknowledgements

The authors are grateful to J. R. Dobbins and Andy Jacob for their careful reading and comments on an earlier version of this paper.

References

[1] GR-246-CORE, Bell Communications Research Specifications of Signaling System Number 7 (SS7), Issue 1. (Bellcore, Dec 1994).

[2] GR-2878-CORE, Generic Requirements for CCS Nodes Supporting High Speed SAAL Links, Issue 1. (Bellcore, Nov 1995).

[3] LATA Switching System Generic Requirements, Section 5, Bellcore, 1994.

[4] ANSI T1.645, B-ISDN Signaling ATM Adaptation Layer - Service Specific Coordination Function for Support of Signaling at the Network Node Interface (SSCF at the NNI), ANSI, 1994.

[5] K. Kant, "Flow Control Mechanisms for SAAL Links" in Broadband Communications, Edited by L. Mason and A. Casaca, Chapman and Hall, 1996, pp 173–184.

[6] D.R. Manfield, G. Millsteed, and M. Zukerman, "Congestion Controls in SS7 Signalling Networks", IEEE Communications Magazine, Vol 31, No 6, June 1993, pp50-57.

[7] "Common Channel Signalling Networks: Performance, Engineering, Protocols and Capacity Management", Special issue of IEEE JSAC, Vol 12, No 3, April 1994.