ATM Networks: Bandwidth Allocation and Congestion Control

M. Gerla, D. Cavendish, and S. Mascolo

Department of Computer Science
University of California, Los Angeles - CA 90024
Phone: (310) 825-4367. Fax: (310) 825-7578 E-mail: gerla@cs.ucla.edu

Abstract - After over 15 years of research and development, prototyping, testbed experimentation and standardization, the ATM (Asynchronous Transfer Mode) technology is finally coming to market. ATM provides a very high speed network infrastructure suitable for local, campus and wide area networks, capable of supporting a broad range of applications, from interactive data to voice and high resolution video. There are still however several areas of ATM which require further work. One such area is *bandwidth allocation* bandwidth allocation. In essence, bandwidth allocation is required in order to guarantee performance to real time traffic connections (eg voice and video). On the one hand, peak bandwidth allocation is inefficient. On the other, "statistical" bandwidth allocation is very difficult, especially in a heterogeneous traffic environment. In this paper, we will introduce a bandwidth allocation technique based on Virtual Private Networks, which can overcome some of these problems. Another critical area in ATM is the *congestion control* of "best effort" traffic, ie traffic which is not allocated any bandwidth, rather, it "fills in" the bandwidth left over by the guaranteed traffic. Hence the name of ABR (Available Bit Rate) traffic. ABR traffic must be flow controlled at the source, to avoid congestion in the ATM network. In this paper, we present a feedback control algorithm for ABR traffic in which source rates are adjusted according to VC queue levels at intermediate nodes along the path. We propose a simple and classical proportional controller, plus a Smith Predictor to overcome instabilities due to large propagation delays. This scheme outperforms PRCA, the congestion control scheme currently considered by the ATM Forum.

0-7803-31-09-5/96/$5.00

I. INTRODUCTION

The goal of an ATM network is to integrate different classes of traffic, making efficient use of resources (bandwidth, buffers, etc) and, at the same time, guaranteeing the Quality of Service (QoS) of each traffic class. Unfortunately, these two requirements (efficient resource utilization and reliable service guarantee) are in conflict with each other. It is easy to satisfy one or the other but not both. The problem of traffic integration and congestion control has been the main focus of ATM research in recent years, leading to various proposals which have been the subject of active debate in the ATM Forum. Currently, the prevailing solution is to subdivide traffic into three classes, CBR, VBR and ABR, and to assign them different priorities at the output trunk.

The Continuous Bit Rate (CBR) traffic includes circuit emulation and fixed rate video. It requires very strict cell loss and delay controls. It is subject to peak policing at the user-to-network interface. It is allocated peak bandwidth in the network, and is transmitted with top priority on the output trunk.

At the other end of the spectrum, the Available Bit Rate (ABR) traffic corresponds to the bulk of "best effort", non real time computer communications traffic (for example, e-mail, file transfers, etc). It can tolerate delay and a limited amount of cell loss (lost cells are retransmitted at the TCP level). It is often presented to the network as "connectionless" traffic, thus precluding the possibility to allocate bandwidth in advance. ABR traffic can be controlled using closed loop, feedback schemes. Currently, the strategy evaluated by the ATM Forum is PRCA (Proportional Rate Control Algorithm) [4]. The PRCA scheme, however, leads to unfair sharing of bandwidth, input rate fluctuations and significant cell loss for connections with large propagation delays. Cell loss causes TCP retransmissions which in turn trigger the intervention of the TCP window control algorithm, with TCP oscillations compounding the PRCA oscillations.

Between CBR and ABR traffic is the very broad class of VBR (Variable Bit Rate) traffic. This is connection oriented traffic with well defined QoS constraints (for example, delay, cell loss rate). The VBR class is extremely diversified, consisting of many different applications (for example, variable rate video, video conferencing, speech, real-time imaging, etc) with different traffic characteristics and QoS requirements. The user must declare the UPC (Usage Parameter Control) parameters for the VBR connection at call set up time (typically, peak rate, average rate and burst length), along with the QoS requirements. Based on UPC and QoS parameters, the network must determine, during the CAC (Call Acceptance Control) phase, if it can accept the call or not. If the call is accepted, bandwidth is allocated to it in a statistical sense (Equivalent Bandwidth allocation concept). Upon accepting the connection, the network will also monitor incoming user traffic to verify that it complies with the initial UPC declaration (policing function).

In addition, the ATM switch must be equipped with sophisticated scheduling disciplines to satisfy the diversified VBR requirements. The typical implementation is a multiple priority structure. CBR traffic gets top priority. Peak bandwidth policing is applied to CBR at the source. Next comes the VBR traffic, with different priority queues corresponding to different QoS requirements. For example, the GDC (General Data Communication) ATM switch supports three VBR queues (high, medium, and low priority). In the Mitsubishi shared memory switch, up to eight different priority classes based on cell loss rate and delay constraints are supported [7]. ABR has lowest priority.

From the above discussion, it is apparent that bandwidth allocation and congestion control of VBR and ABR traffic are critical to the efficient operation of the ATM network. In this paper, we present solutions which help alleviate this problem. First, we propose a Virtual Private Network technique for VBR traffic. Then, we introduce an enhanced PRCA scheme, called SP-PRCA, for ABR traffic. We conclude the paper by outlining the problems which still remain to be attacked.

II. ATM VIRTUAL PRIVATE NETWORKS

From the preceding section, it is apparent that in a public ATM network which allows *unrestricted* sharing of bandwidth among heterogeneous VBR users, it is difficult to reconcile the goals of efficient resource utilization and QoS guarantee. This is because the accurate and consistent mapping of different types of traffic into different QoS requirements to a limited number of queues is a very challenging task. One way of alleviating this problem is to **partition** bandwidth among groups of users with similar characteristics. In other words, we give up the potential advantages of complete, unrestricted sharing , for the sake of a more manageable bandwidth allocation. In this section, we pursue such an approach. More precisely, we propose to identify user groups by some well defined criterion (e.g. enterprise, traffic class) and assign them to different Virtual Private Networks (VPNs). The VPN is a virtual net embedded in the original ATM WAN. The nodes of the VPN are represented by a subset of the ATM WAN switches. The links of the VPN are VPs established between ATM switches. These VP links are allocated peak bandwidth at VPN initialization. In essence, the VPN user sees a network with dedicated bandwidth, in which he can establish and manage his own VCs.

The easiest way to build a VPN is to use End-to-End VPs. A user with a number of sites can interconnect them with a completely connected VP mesh. Fig. 1 shows an example of such a network. The user site A establishes end-to-end VPs with the other sites. All the traffic destined for sites B, C, D and E are routed on VPs AB, AC, AD and AE, respectively. The other user sites can also establish VPs between one another. The ATM network allocates *peak bandwidth* to VPs and performs peak bandwidth policing on each VP accordingly. The user can multiplex several VCs within each VP.

This scheme makes VPN design extremely simple. A serious drawback of the EEVP strategy, however, is bandwidth fragmentation over many VPs. This prevents statistical multiplexing of traffic between different source-destination pairs and dynamic adjustment to changes in traffic pattern. Another drawback of this scheme is that it precludes scaling to a large number of sites. As the number of nodes in the network increase, the number of VPs required to interconnect them also increase, aggravating the problem of bandwidth fragmentation. Further, the VP tables at the VP switches become large making switching more expensive.

These problems can be efficiently solved by constructing a Broadband (or mesh) Virtual Private Net-

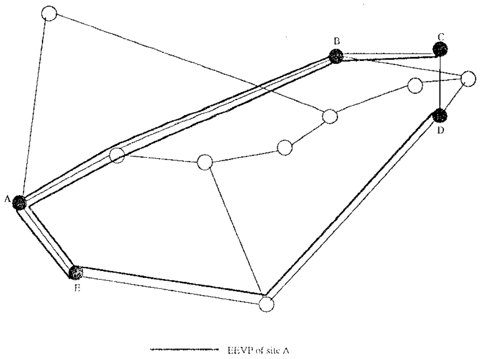


Fig. 1. End-to-end VPs

work [8]. The Broadband VPN is a mesh of VP links connecting ATM nodes to one another and to user sites. Fig. 2 shows a typical example of Broadband VPN. As in the EEVP strategy, VPs are allocated *peak bandwidth*. Also, a logical mesh of end-to-end VCs is defined. However, a VC is not confined to a single end-to-end VP, but may traverse several VPs. For example, the VC from site A to site C is switched from VP AB to VP BC at site B. Thus, network nodes provide not only VP switching, but also VC switching.

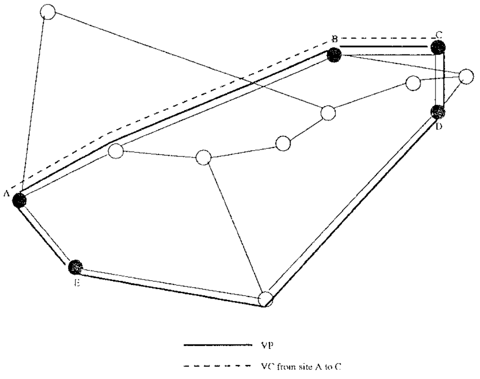


Fig. 2. Broadband Virtual Private Network

The private customer network has a Local Customer Manager (LCM) at each site and, optionally, a Central Customer Manager (CCM) to coordinate the operations of the LCMs, as shown in Fig. 3. CCM and LCMs cooperate to carry out network management functions, such as CAC (Call Acceptance Control), routing, bandwidth reallocation and topology reconfiguration.

The VP mesh solution offers considerable advantages over the EEVP strategy. First, traffic streams between different source/destination sites can statistically share a common VP when their paths overlap. Second, traffic pattern changes are more easily absorbed by a well designed VP topology, without requiring dynamic reallocation of bandwidth.

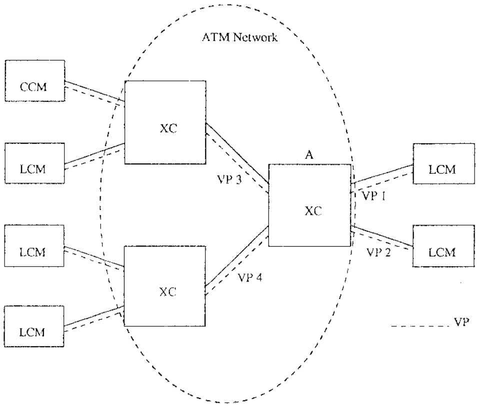


Fig. 3. BVPN Network Management

These advantages, hoewver, do not come free. For example, the ATM node must cross connect now not only VPs, but also VCs. So, VC routing maps must also be maintained. Furthermore, unlike the EEVPN case, congestion can occur in a mesh VPN at intermediate ATM XCs(cross connects). For instance, consider the ATM XC "A" in Fig. 3. Assume that initially the traffic into and out of node A is balanced, and therefore bandwidth allocation is the same for VP 1, 2, 3 and 4. Accordingly, the input VP policing parameters are set to the same peak value for all VPs. Because of traffic fluctuations, it is possible that during a certain interval all traffic entering from VP 1 and VP 2 goes out to VP 4. Clearly this overload will cause congestion in VP 4. In the sequel, we review two strategies, Fast Resource Management (FRM) and Intra-Node Policing (INP), to solve the problem of internal congestion in an ATM XC.

III. BVPN CONGESTION CONTROL

In the FRM strategy the user requests peak bandwidth allocation for each incoming burst of traffic or connection [5]. When a burst arrives at the source , a fast reservation cell is issued on the proper VC. The first ATM XC on the path verifies that the request can be accepted on the incoming and outgoing VPs (i.e., it keeps track of user bandwidth allocation on VP links). Then, it forwards the fast reservation request to the next node along the path, which in turn, verifies that the request can be accepted on the next VP. If any one of the intermediate ATM XC does not have enough bandwidth available for the requesting burst, a Negative ACK (NACK) is sent back to the source.

An alternative congestion control technique is intra-node policing. To this end, consider Fig. 4 and assume that the input traffic from VP 1 is split into rates $R13$ and $R14$ directed to VP 3 and VP 4 respectively. Similarly, VP 2 is split into $R23$ and $R24$.

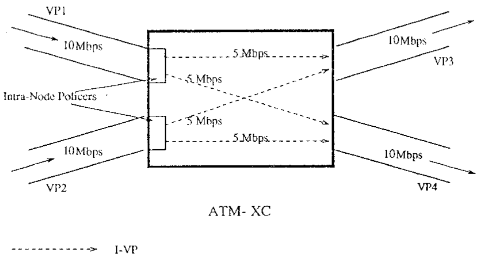


Fig. 4. Intra-Node Policing example

The condition to avoid ATM XC congestion is that $R13 + R23 <$ peak bandwidth allocated to VP 3. Ideally, we would like to jointly police the sum $R13$ and $R23$. Unfortunately, this is not possible since these two traffic streams enter from two different different trunk interfaces. Policing requires counting cells in real time, and therefore it can be performed only on the trunk card on which the traffic stream is entering. A more realistic solution entails the policing of $R13$ and $R23$ separately. This policing strategy, which we call Intra-Node Policing (INP) is rather easy to implement. It simply requires keeping a common policing counter for all the VCs which are routed from VP 1 to VP 3. If $R13$ and $R23$ are chosen so that $R13 + R23 <$ peak bandwidth allocated on VP 3, nodal congestion is prevented. Thus the scheme is fail-safe. In Fig. 4 example, the VP1 traffic is policed down to 5Mbps for each of the two subroutines routed to VP3 and VP4, respectively.

A quantitative comparison of FRM, INP and EEVP schemes was carried out on the mesh topology shown in Fig. 3. Fig. 5 shows the amount of bandwidth required by the network to meet burst loss probability of 10^{-4}. FRM and INP schemes perform equally well. EEVP performs poorly compared to the other two due to lack of statistical multiplexing that is achievable with FRM and INP schemes.

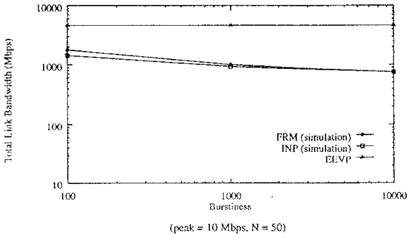


Fig. 5. Link bw vs burstiness in NSF topology

In summary, the VPN approach offers the following benefits:

- simplified bandwidth allocation. The network provider does peak bandwidth allocation and policing. The user is responsible for statistical bandwidth allocation among VCs within his VPN.
- customized QoS is easier in a VPN, since traffic is homogeneous and traffic characteristics are well known to local and central customer managers.
- the user has more direct control over ATM resources.
- peak bandwidth allocation on VPs drastically reduces interference between different VPNs.
- The VPN user has still the option to access all the conventional ATM WAN services (in addition to VPN); i.e. he/she can establish VC connections outside of VPN to communicate with non VPN users.

IV. ABR CONGESTION AVOIDANCE

We start by giving a brief description of the ABR feedback control. Various VC's can share common links in the network, giving rise to congestion. In order to avoid cell loss, sources sharing a congested link regulate their input rates based on a feedback mechanism. We assume that the source of a VC sends a Resource Managment (RM) cell every NRM (non RM) data cells. The RM cell is relayed back to the source by its destination node, containing some feedback information. This scheme is called Forward Congestion Notification (FCN) and is recommended by ATM Forum. Along a VC, there exists only one bottleneck queue, which may change position according to traffic conditions. By controlling the level of the bottleneck queue, we avoid overflow at each and every queue along the path. More precisely, the RM cell, travelling along the VC, gets stamped with the maximum queue level. Fig. 6 shows the block diagram model of the system consisting of the traffic source, the controller, the forward delay (from source to bottleneck), the feedback delay (from bottleneck to source) and the bottleneck queue.

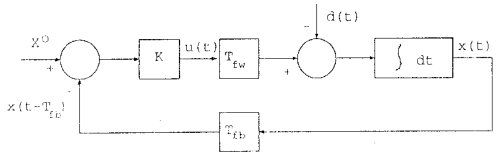


Fig. 6. Queue model of a proportional controlled source

Note that in this scheme the control is performed at the input (it is called input rate control). The controller is a simple proportional controller plus a Smith Predictor to ovecome instabilities due to long propagation delays. The controller is described in details in [10]. In the following we describe only how the resulting control algorithm operates at each input source. The source calculates the difference $(X^o - (X(t - t_{fb}) + \int_{t-RTD}^{t} u(t^{'})dt^{'})$, where X^o is the set threshold level of the queue, $x(t - t_{fb})$ is the bottleneck queue level relayed at t_{fb} units of time ago, and the integral is the amount of cells pumped into the network by the source during the last round trip delay. The rate computed at the source is proportional to this difference, i.e.:

$$rate = u(t) = K[X^o - x(t) - \int_{t-RTD}^{t} u(t^{'})dt^{'}] \quad (1)$$

We can see that this algorithm considers all the "in flight cells" as if they were in the bottleneck, i.e. it considers "in pipe cells" like a pseudo queue added to the bottleneck. Intuitively this can be seen as a way to overcome the problem of the delayed feedback information $x(t - t_{fb})$. The resulting behavior of the bottleneck queue level is stable, cell loss free and without oscillations.

In the ATM Forum PRCA proposal [11], an additive increase/multiplicative decrease rate control is exercised at the sources. Namely, binary feedback information (congested/ not congested) is received at the sources, and rate increase (additive) is performed in case a "not congested" feedback is received. Failure to receive the "not congested" notification causes multiplicative rate decrease at the source after each time interval Δ, thus making the scheme conservative.

In contrast, our proposed EPRCA (Enhanced PRCA) uses the delayed queue occupancy as the feedback information. Like in the PRCA scheme, if no feedback is received, the source calculates the rate at fixed intervals Δ related to the time constant of the queue. As mentioned before, the calculation is performed using a "worst case" estimate of the queue level.

The EPRCA scheme, based on a simple mathematical model, operates according to a "positive feedback" mechanism, much like the PRCA scheme. The important difference is that the dynamic behavior of our regulation is related to the network state and parameters. In fact, the rate decreases exponentially with a base related to the sampling time/time constant ratio. More importantly, the increasing jumps are related to the queue level and to the number of cells released from the source during the last round trip interval. As a consequence, our EPRCA scheme does not drop cells, nor does it need a queue size proportional to the round trip delay to prevent cell loss. In contrast, the conventional PRCA scheme does not use precise information on the queue level and does not take into account the number of cells released during the last round trip delay. Consequently, it cannot perform the correct rate increase so as to prevent congestion and cell loss.

We have simulated both the PRCA and EPRCA in the network shown in Fig. 7. Links have uniform speeds, normalized to 1 cell per unit of time [cell/s]. Links to the right of the bottleneck have a bandwidth-delay product of 10 cells, while the links to the left of the bottleneck have zero propagation delay.

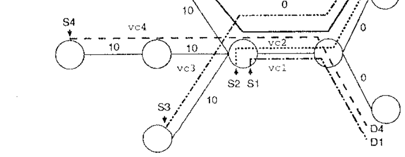


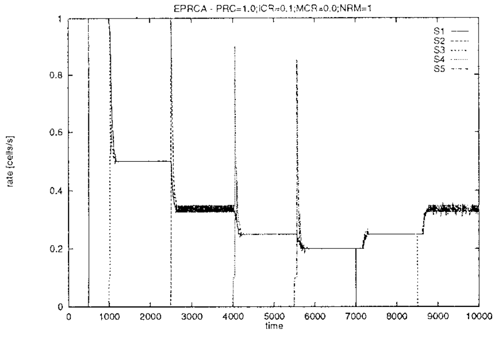
Fig. 7. Network Topology

Five VC connections compete for bandwidth resources of a bottleneck link. VC connection activity (i.e. start and end time) is described in Table 1.

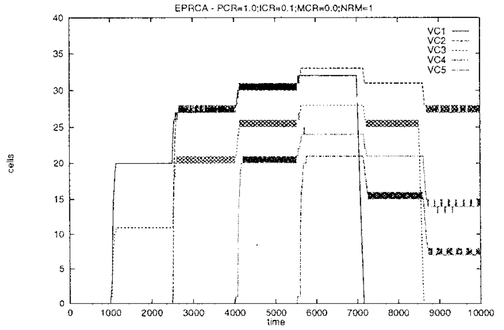
Table 1: VC Connections Activity

Connection #	1	2	3	4	5
Start Time	500	2500	1000	4000	5500
End Time	7000	10000	8500	10000	10000

Fig. 8(a) shows the behavior of the five input rates, corresponding to connections $S1-S5$, at source nodes. For sake of comparison with PRCA, we assume an initial cell rate of 0.1 [cells/s], equal to the PRCA minimum cell rate. After the start/end of a connection, each rate rapidely settles on the new fair stationary value. Fig. 8(b) shows the dynamic behavior of the five queues at the bottleneck link, corresponding to $VC1 - VC5$ bottleneck queues. As can be seen, no queue overflow occurs.



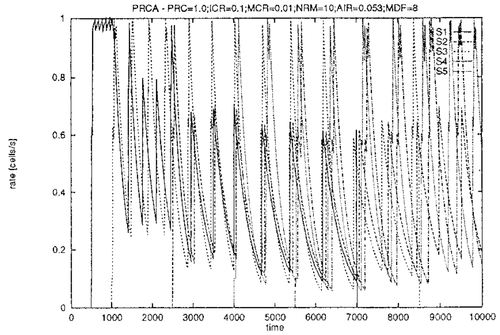
(a) Rates



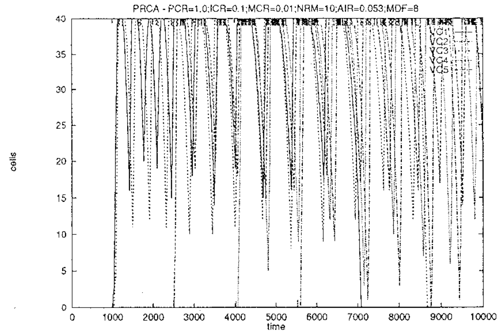
(b) Bottleneck Queues

Fig. 8. Periodic Feedback

The PRCA scheme has been simulated under the same traffic conditions as before, with parameters: NRM=10; AIR=0.053; MDF=8. The results are shown in Fig. 9. As expected, the PRCA scheme does not prevent cell loss, because it cannot account for the bottleneck queue level and the number of cells "in flight".



(a) Rates



(b) Bottleneck Queues

Fig. 9. PRCA Control Scheme

V. CONCLUSIONS

Statistical bandwidth allocation for VBR traffic and congestion control for ABR traffic have emerged as the major challenges in ATM. In this paper we have proposed solutions for both problems. In particular, we have shown that the VBR bandwidth allocation. Problems can be alleviated by allowing specific classes of users to reserve bandwidth with a strategy known as VPN. The network provider only performs peak policing on VPN traffic. The responsibility of efficiently utilizing the reserved VPN bandwidth and complying with QoS requirements is now shifted to the VPN customer management. In the VPN area, work is now in progress on the development of an ATM network control and management platform which allow. VPN managers to efficiently control their networks. This includes the ability to monitor VPN bandwidth allocation and user traffic, and to dynamically reconfigure virtual topology and capacities based on demands.

For Best Effort unicast (point to point ABR) traffic, we have shown an effective congestion control algorithm that outperforms the PRCA scheme. Work is in progress to extend this algorithm to multicast traffic.

REFERENCES

[1] Comer, "Internetworking with TCP/IP," volume 1, *Prentice Hall*,1991.

[2] Crocetti, Fotedar, Fratta, Gallassi, and Gerla, "ATM Virtual Private Networks Design Alternatives", *Computer Communications*,18(1),pp. 24-31, Jan. 1995.

[3] Gallassi et al., "ATM : Bandwidth Assignment and Bandwidth Enforcement Policies." *IEEE Globecom '89* , 1989.

[4] Golestani. "A framing strategy for congestion management." *IEEE JSAC*, 9(7):1064-1077, September 1991.

[5] Kung et al. "Use of link by link Flow Control in Maximizing ATM network performance: simulation results." In *IEEE Hot Interconnects Symposium '93*, August 1993.

[6] Leland, Taqqu, Willinger, and Wilson. "On the Self-Similar Nature of Ethernet Traffic." *IEEE/ACM Trans. on Networking*, 2(1):1–15, Feb 1994.

[7] Monteiro et al., "Leaky Bucket Input Rate Control in ATM Networks", In *ICC*, 1990.

[8] Walters and Ahmed, "Broadband Virtual Private Networks and Their Evolution", In *XIV ISS Proceedings, Yokohama*, October 1992.

[9] Yamanaka et al., "622 Mbps 8X8 Shared Multibuffer ATM Switch with hierarchical queueing and Multicast Functions", In *IEEE Globecom '93*, December 1993.

[10] S. Mascolo, D. Cavendish, M. Gerla,"ATM Rate Based Congestion Control Using a Smith Predictor: an EPRCA Implementation", *Proceedings of IEEE INFOCOM'96*.

[11] A. W. Barnhart, "Baseline Performance Using PRCA Rate-Control," *ATM Forum/94-0597*, July 1994.