ATM Switching

Thomas M. Chen
Dept. of Electrical Engineering
Southern Methodist University
PO Box 750338
Dallas, TX 75275-0338
Tel: 214-768-8541
Fax: 214-768-3573
Email: tchen@engr.smu.edu

Stephen S. Liu
Verizon Laboratories
40 Sylvan Road
Waltham, MA 02451
Tel: 781-466-2809
Email: steve.liu@verizon.com

Abstract: ATM switches are high-speed packet switches designed to process and forward ATM cells. The functional components of a general ATM switch architecture are described. The switch fabric provides the central function of cell forwarding. The switch fabric design involves trade-offs between various approaches and a choice of input or output buffering. In addition to cell forwarding, ATM switches require capabilities for connection admission control, traffic control, and OAM (operations and maintenance).
Keywords: ATM, cells, switching, connection control, traffic control, buffering

1. Introduction

ATM (asynchronous transfer mode) is an internationally standardized connection-oriented packet switching protocol designed to support a wide variety of data, voice, and video services in public and private broadband networks (1,2). ATM networks generally consist of ATM switches interconnected by high-speed transmission links. ATM switches are high-speed packet switches specialized to process and forward ATM cells (packets). Since ATM is a connection-oriented protocol, ATM switches must establish a virtual connection from one of its input ports to an output port before forwarding incoming ATM cells along that virtual connection.

An ATM cell consists of a 5-byte header followed by a 48-byte information field or payload. The main purpose of the ATM cell header is to identify the virtual connection of the cell which occupies most of the header bits. An ATM virtual connection is specified by the combination of a 12-bit virtual path identifier (except the first 4 bits are used for generic flow control at the user-network interface) and a 16-bit virtual channel identifier. Virtual paths are bundles of virtual channels. VP crossconnects are designed to route ATM traffic on the basis of virtual paths only, which is convenient when large amounts of traffic must be routed or rerouted at the same time. The VPI/VCI fields are followed by a 3-bit payload type (PT), 1-bit cell loss priority (CLP), and 8-bit header error control (HEC) field. The PT field is used to distinguish control cells from data cells, and explicit forward congestion indication (EFCI). The CLP flag is used to indicate that lower priority (CLP=1) cells should be discarded before CLP=0 cells in the event of congestion. The HEC field allows single bit-error correction and multiple bit-error detection over the cell header.

A generic ATM switch architecture with N input ports and N output ports is shown in Figure 1 (note switches can have any dimensions). The functions of an ATM switching system may be divided broadly into user cell forwarding, connection control, and network management (3). ATM cells containing user data are received at the input ports, and the input port processors prepare the cells for routing through the switch fabric. The fabric in the center of the switching system provides the interconnections between input port processors and output port processors. The output port processors prepare the outgoing user cells for transmission from the switch. User cell forwarding is characterized by parallelism and high-speed hardware processing. The ATM protocol was intentionally streamlined to allow incoming cells to be processed simultaneously in hardware and routed through the switch fabric in parallel. Thus, ATM switches have been able to realize high-end performance in terms of throughput and cell forwarding delay.

Connection control, sometimes called the control plane, refers to the functions related to the establishment and termination of ATM virtual connections. Connection control functions generally encompass: exchange and processing of signaling information; participation in routing protocols; and decisions on admission or rejection of new connection requests.

Network management is currently carried out by SNMP (simple network management protocol), the standard protocol for managing data networks. ATM switches typically support an SNMP agent and an ATM MIB (management information base). The SNMP agent responds to requests from a network manager to report status and performance data maintained in the MIB. The agent might also send alarms to the network manager when prespecified conditions are detected. Since ATM switches can be viewed as a specific type of network element covered within the SNMP framework, the

details of SNMP functions in ATM switches are not discussed in detail here.

Network management should also include standardized ATM-layer OAM (operations and maintenance) functions. ATM switches carry out OAM procedures by generating, exchanging, processing, and terminating OAM cells. OAM cells are used for fault management, performance management, and possibly other ATM-layer management functions.

2. Input and Output Port Processing

The input port processors carry out several important functions. First, the physical layer signal is terminated. For the common case of SONET/SDH (synchronous optical network/synchronous digital hierarchy), the SONET/SDH framing overhead fields are processed, and the payload is extracted from the frame. Individual 53-byte ATM cells are delineated in the payload.

Next, each cell header undergoes a number of processing steps. The cell header is checked for bit errors using the HEC field, and cells with uncorrectable header errors are discarded. The traffic rate of each virtual path or virtual channel is monitored according to an algorithm called the generic cell rate algorithm or GCRA, which is essentially a leaky bucket algorithm. A switch may be configured to allow, discard, or "tag" cells (by setting CLP=1) exceeding the allowed traffic rate. The VPI/VCI value in each cell header is used to index a routing table to determine the proper output port and outgoing VPI/VCI values. Incoming VPI/VCI values must be translated to outgoing VPI/VCI values by every ATM switch. Cells requiring special handling, such as signaling cells and OAM cells, must be recognized and routed to the appropriate processors in the switch. User cells are prepared for routing through the switch fabric, often by prefixing a routing tag to the cell. The routing tag may consist of the output port, service priority, type of cell, timestamp, or other information for routing and housekeeping purposes. Since the routing tag exists only within the switch, its contents may be chosen entirely by the switch designer. Before entering the switch fabric, cells may be queued in a buffer in the input port processor.

The output port processors have the opposite role of the input port processors, namely preparing ATM cells for physical transmission from the switch. Cells from the switch fabric may be queued in a buffer in the output port processor, in which case the switch is called an output buffered switch. If routing tags are used, the output port processors remove the routing tag from each user cell. If special cells, such as signaling cells and OAM cells, need to be transmitted, they are inserted into the outgoing cell stream. A new HEC field is calculated and inserted into each cell header. Finally, the ATM cells are transmitted as a physical layer signal. In the case of SONET/SDH, cells are mapped into the payloads of SONET/SDH frames.

3. Switch Fabrics

It is often convenient to visualize the basic operation of an NxN switch synchronized to periodic time intervals equal to the transmission time of one cell, referred to as a "cell time," assuming that the transmission rate on all links are equal. For example, the cell time for a 155-Mb/s transmission link would be approximately 53 bytes/155 Mb/s = 2.7 μs. In each cell time, a new set of N incoming cells may appear at the input ports and up to N outgoing cells may depart from the output ports. The N incoming cells are processed in parallel by the input port processors and presented simultaneously to the switch fabric. The switch fabric attempts to route the cells in parallel to their appropriate output ports.

There is a chance that more than one cell may attempt to reach the same output port at the same time, called output contention, which has four significant consequences. First, one cell may reach the output port but the other cells would be lost without buffers existing somewhere in the switch to temporarily store these cells. Switch fabric designs differ in their choice of buffer placement. Second, queues may accumulate in the buffers resulting in random cell delay and cell delay variation, which are usually two performance metrics of interest. Third, buffers are necessarily finite implying the possibility of buffer overflow and cell loss. Some fabric designs may not be able to handle a full traffic load without a probability of cell loss. A performance metric for switch fabrics is the normalized throughput or utilization defined as the overall fraction of a full traffic load that can be forwarded successfully through the fabric. Ideally, switch fabrics should be capable of 100 percent utilization. Fourth, some switch fabrics must operate at a rate faster than the transmission link rate. The ratio of the switch fabric rate to the transmission link rate is sometimes referred to as a "speedup factor." A speedup factor is often related to the scalability of a switch fabric design, that is, the difficulty of constructing an arbitrarily large fabric (3).

3.1. Shared Memory and Shared Medium

A speedup factor of N is evident in switch fabric designs based on a shared memory or shared medium, which are shown in Figure 2. In a shared memory design, incoming cells are first converted from serial to parallel form. They are written sequentially into a dual port random access memory (4). Their cell headers with routing tags are directed to a memory controller that keeps track of the memory location of all cells associated with each output port. The memory controller links the memory location of outgoing cells to maintain

virtual output queues. The outgoing cells are read out of the memory, demultiplexed, and converted from parallel to serial form for delivery to the output port processors. Since the cells must be written into and read out from the memory one at a time, the shared memory must operate at the total throughput rate. Hence, it must be capable of writing N cells and reading N cells in one cell time, implying a speedup factor of N. As a consequence, the size of the fabric, N, will be limited by the memory access time. On the other hand, a shared memory design has been popular due to its simplicity and efficient sharing of memory space.

Similarly, cells may be passed from input port processors to output port processors through a high-speed time division multiplexed (TDM) bus. Incoming cells are sequentially broadcast on the bus. At each output, address filters examine the routing tag on each cell to determine if the cell is addressed for that output. The address filter passes the appropriate cells through to an output buffer. Shared bus designs have been used in traditional router architectures due to their simplicity and modularity. On the other hand, the buffer space is not shared as efficiently as a shared memory. Also, the bus speed must be fast enough to carry up to N cells in each cell time, corresponding to a speedup factor of N. The address filters and output queues must operate at the bus speed as well. The size of a shared bus fabric will be limited by the expense and complexity of high-speed hardware for the bus, address filters, and output queues.

3.2. Space Division

A simple example of a space division fabric is a crossbar switch shown in Figure 3 which was originally developed for telephone switching. An NxN matrix of crosspoints can connect any of the N inputs to any of the N outputs. While a crossbar switch has the advantages of simplicity and no speedup factor, it has two major disadvantages. First, a crossbar switch will have output blocking, meaning that only one cell may be delivered to an output port at a time. Other cells contending for the same output port may be queued at the input ports, but the normalized throughput for an input buffered fabric is well known to be only $2 - 2^{1/2} = 0.586$ for large N, assuming uniform random traffic, i.e., an incoming cell attempts to go to any output port with equal probability independent of all other conditions (5).

Second, the N^2 number of crosspoints does not scale well to large fabrics. As an alternative, multistage interconnection networks (MINs) have been studied extensively over many years of development of telephone switches (6). MINs are constructed by connecting a number of small switching elements, often 2x2 switching elements, in a regular pattern. Banyan networks are a popular class of MINs used for ATM switch fabrics. Figure 4 shows an example of an 8x8 banyan network. The dashed outlines emphasize that the 8x8 banyan network is constructed by adding a third stage to interconnect 4x4 banyan networks, which are in turn constructed by an interconnection of two stages of 2x2 switching elements. An n-level banyan may be constructed by connecting several (n-1)-level banyans with an additional stage of switching elements. This recursive and modular construction of larger fabrics is a significant advantage for implementation.

Another advantage is the simplicity of the 2x2 switching elements. Each 2x2 switching element routes a cell according to a control bit. If the control bit is 0, the cell is routed to the upper output (address 0); otherwise, the cell is routed to the lower output (address 1). Delta networks are a subclass of banyan networks with the "self-routing" property: the output address of a cell also controls the route of that cell. For example, the cell shown in Figure 4 is addressed to output port 010. The n-th bit of the address "010" is used as the control bit in the n-th stage to route the cell to the proper output port, and this self-routing works regardless of which input port the cell starts from. The self-routing property simplifies the control of the delta network switch fabric.

Delta networks can take different forms, depending on their method of construction, including omega, flip, cube, shuffle-exchange, and baseline networks (7). A delta network of size NxN constructed of MxM switching elements will have $\log_M N$ stages, each stage consisting of N/M switching elements. Hence, if M=2, the total number of crosspoints will be on the order of $N \log_2 N$ which compares favorably to N^2 crosspoints in a crossbar switch.

Unfortunately, the savings in number of crosspoints comes at the cost of possible internal blocking, meaning that the routes of two cells addressed to different outputs might conflict for the same internal link in the fabric before the last stage. In this situation, only one of the two cells for a link can be passed to the next stage, while the other cell stays behind, queued either in a buffer within each switching element or in an input buffer. Thus, internal blocking will cause a loss of throughput. A well-known solution is to add a Batcher sort network to rearrange the cells according to an increasing or decreasing order of addresses before the banyan network (8). A combined Batcher-banyan network will be internally nonblocking in that a set of N cells addressed to N different outputs will not cause an internal conflict. However, output blocking can still occur if two cells are addressed to the same output, and it must be resolved by buffering.

An obvious possibility is input buffering before the Batcher-banyan network. If more than one cell is addressed to the same output, one cell is allowed to pass through the Batcher-banyan network while the other cells remain in the input buffers. Naturally, throughput will be lost due to the so-called head-of-line blocking,

where a delayed cell prevents the other cells waiting behind it from going through the fabric. Many approaches are possible to overcome the head-of-line blocking problem and increase the throughput of the fabric, such as: increasing the speedup factor; distributing the traffic load to multiple banyan networks in parallel; cascading multiple banyan networks in tandem; or virtual output queueing where N separate virtual queues corresponding to the output ports are maintained at each input port. Although these solutions add complexity to the fabric implementation, space-division fabrics are still attractive for their ability to scale to large sizes. Large fabrics may be constructed as MINs composed of small switching modules, where the small switching modules can be any type of fabric design.

3.3. Input and Output Buffering

The placement of buffers in the switch can have a significant effect on the switch performance. Fortunately, this issue has been studied extensively. Figure 5 shows three basic examples: input buffering, output buffering, and internal buffering. Input buffering is known to suffer from head-of-line blocking without special provisions to overcome it. Output buffering is generally agreed to be optimal in terms of throughput and delay (5). However, output buffering often involves a speedup factor which limits the scalability to large fabrics.

The addition of buffers within the switching elements of a banyan network to resolve internal blocking has not been shown to improve the throughput substantially. An interesting fabric design is a crossbar switch with buffers at each crosspoint (9). Incoming cells are dropped into the appropriate buffer corresponding to the output. Each output multiplexes the cells queued in N buffers. The buffered crossbar switch (also called bus-matrix switch) is actually an output buffered fabric as illustrated in Figure 6. It offers the desirable performance of output buffering with no speedup factor. However, it does not share buffer space efficiently, and the number of output buffers scales exponentially as N^2. The Knockout switch shown in Figure 6 reduces the number of output buffers to NL where L is a constant by the addition of N:L concentrators at each output (10). It has been noted that under uniform random traffic conditions, the probability of more than L cells addressed to the same output port in the same cell time will be very small if L is chosen appropriately large. The N:L concentrators allow up to only L cells to pass in one cell time to L output buffers at each output; additional cells are lost. However, if L is chosen to be 8 or greater, the cell loss ratio will be 10^{-6} or less. At the cost of a small cell loss, the scalability of the fabric becomes linear with N instead of exponential.

4. Connection Control

Since ATM is a connection-oriented protocol, virtual connections must be established before any user cells can be forwarded. Virtual connections may be permanent, semi-permanent controlled through network management, or dynamically established by means of ATM signaling in response to user requests. ATM switches exchange signaling messages along a selected route and make decisions about allocation of switch resources to new user requests. Usually route selection is carried out by a separate process. Routes may be static or dynamically chosen through a routing protocol. The PNNI (private network node interface) routing protocol is a dynamic link-state routing protocol similar to the OSPF (open shortest path first) protocol used in the Internet.

4.1. Signaling

ATM switches must participate in signaling protocols, either access signaling between the user and edge switch or interoffice signaling between two switches. The ATM access signaling protocol is the ITU-T standard Q.2931 which was derived from the ISDN access signaling protocol Q.931. Q.2931 signaling messages are encapsulated in ATM cells using a signaling ATM adaptation layer (SAAL) protocol. Signaling cells are exchanged on a pre-established signaling virtual channel (VCI=5) or another signaling virtual channel dynamically established through meta-signaling (a pre-established meta-signaling virtual channel identified by VCI=1).

The high-layer interoffice signaling protocol is the ITU-T standard BISUP (broadband ISDN user part) derived from the ISDN user part of Signaling System 7 (SS7). BISUP messages may be exchanged directly between ATM switches, where BISUP messages would be encapsulated into ATM cells using SAAL, or sent through the existing SS7 packet-switched network.

Figure 7 shows a typical exchange of signaling messages between ATM switches to successfully establish and release a virtual connection. Basically, a Q.2931 “setup” message is first sent by the user to request a new virtual connection. If each switch decides to accept the request, a BISUP “initial address” message (IAM) is forwarded along a selected route. The IAM message includes all information required to route the connection request to the destination user, such as destination user address, service class, ATM traffic descriptor, connection identifier, quality of service (QoS) parameters, and additional optional parameters. A Q.2931 “setup” message notifies the destination user. If the connection is accepted, a series of signaling messages are returned in the reverse direction to alert the calling party that the connection is established. The reverse signaling messages also serve to finalize the resource reservations that were made earlier tentatively

in each switch. When the virtual connection is no longer needed, a "release" message will free the reserved resources at each switch to be used for another connection.

The complete ATM signaling protocol, including additional signaling messages, options, and timing requirements, is elaborate to implement. Obviously, signaling cells require special processing within the switch. Incoming signaling cells are recognized and diverted to a signaling protocol engine for processing. Outgoing signaling cells from the signaling protocol engine are multiplexed into the outgoing cell streams.

4.2. Connection Admission Control

ATM supports the notion that accepted virtual connections will be guaranteed their requested level of QoS – mainly in terms of maximum cell delay, cell delay variation, and cell loss ratio – or otherwise, a new connection request should be rejected. Hence, the acceptance of a virtual connection is an implicit agreement between the user and network on a mutual understanding of their respective obligations, often called a "traffic contract." The user side of the traffic contract involves conformance to the ATM traffic descriptor or traffic rate parameters. The network side of the traffic contract is a guarantee of the requested QoS for the conforming traffic.

Naturally, not every connection request may be accepted because network resources are shared for higher efficiency. If too much traffic is admitted, the QoS for existing connections may deteriorate below their guaranteed levels. On the other hand, the network should attempt to accept as much traffic as possible to maximize efficiency and revenue. Connection admission control refers to the general process for deciding acceptance or rejection of new connection requests. The main issue for an ATM switch is whether sufficient resources are available to satisfy the QoS requirements of the new connection and all existing connections. Because ATM traffic is random, the effect of a new connection cannot be known precisely during CAC. The switch follows a CAC algorithm chosen by the network provider to estimate the impact of a new connection.

The numerous CAC algorithms studied over the years can be broadly classified as deterministic or statastical. Deterministic approaches calculate the effect of a new connection based on a deterministic traffic envelope characterizing a bound on the shape of the expected traffic, e.g., peak cell rate or a leaky bucket-limited envelope. Statistical methods usually estimate the effect of a new connection by carrying out a stochastic analysis of a queueing model. Statistical methods can be classified as model-based or measurement-based (or a combination of both). Model-based approaches make an assumption about traffic models as inputs to a queueing model. Measurement-based approaches depend on measurements of actual traffic as inputs to an analytical model. In any case, the CAC algorithm is not a matter for standardization and should be chosen by the network provider.

4.3. Routing

The ATM protocol is not tied to a specific routing protocol. Indeed, a dynamic routing protocol is not needed if routes are static. Also, the concept of semi-permanent virtual paths was intentionally included in ATM to simplify the routing process. Virtual paths can serve as large "pipes" with allocated bandwidth between pairs of nodes. If a new connection finds a convenient virtual path to its destination, it can make use of an available virtual channel within that virtual path with minimal set-up overhead at intermediate switches.

For dynamic routes, PNNI routing is a link-state routing protocol. ATM switches will periodically advertise information about its links and maintain a topological view of the network constructed from link-state advertisements from other switches. These functions are carried out by a routing protocol engine within the connection control function.

5. Traffic Control Considerations

ATM switches are responsible for a comprehensive set of traffic control mechanisms to support QoS guarantees in addition to connection control (11,12). For the most part, these other mechanisms operate in various parts of the switch independently of connection control.

5.1. Usage Parameter Control

Although the source traffic is expected to conform to the traffic descriptor negotiated during connection establishment, the actual source traffic may be excessive for various reasons. To protect the QoS of other connections, the source traffic rate needs to be monitored and regulated at the user-network interface by ATM edge switches. Usage parameter control (UPC) is the process for traffic regulation or "policing" carried out by a leaky-bucket algorithm called the generic cell-rate algorithm. The generic cell rate algorithm involves two parameters, an increment I and a limit L, and is therefore denoted as GCRA(I,L). The parameter I is inversely proportional to the average rate allowed by the GCRA while the parameter L determines its strictness. The GCRA is activated for a virtual connection after it has been accepted.

The operation of the GCRA is illustrated in Figure 8. A bucket of capacity I+L drains continuously at a rate of 1 per unit time. A cell is deemed to be conforming if the bucket contents can be incremented by I without overflowing; otherwise, the cell is deemed to be non-conforming or excessive. Conforming cells should be

allowed to pass the GCRA without any effect. The network administrator can choose non-conforming cells to be allowed, discarded, or tagged by setting CLP=1 in the cell header.

A virtual scheduling algorithm offers an alternative but equivalent view of the GCRA. The actual arrival time of the n-th cell, t(n), is compared with its theoretical arrival time T(n), which is the expected arrival time assuming that all cells are spaced equally in time with separation I. Cells should not arrive much earlier than their theoretical arrival times, with some tolerance dependent on L. A cell is deemed to be conforming if $t(n) > T(n) - L$; otherwise, it is non-conforming (too early). The theoretical arrival time for the next cell, T(n+1), is calculated as a function of t(n). If the n-th cell is conforming and $t(n) < T(n)$, then the next theoretical arrival time is set to $T(n+1) = T(n) + I$. If the n-th cell is conforming and $t(n) \geq T(n)$, then the next theoretical arrival time is $T(n+1) = t(n) + I$. Non-conforming cells are not counted in the update of the theoretical arrival times.

Multiple GCRAs may be used in combination to regulate different sets of parameters. For example, a dual leaky bucket may consist of a GCRA to regulate the peak cell rate followed by a second GCRA to regulate the sustainable cell rate (an upper bound on the average rate). A conforming cell must be deemed conforming by both GCRAs.

5.2. Packet Scheduling

ATM does not allow indication of service priority on the basis of individual cells. Although service priorities can be associated with virtual connections, it is common to group virtual connections according to their class of service, such as real-time constant bit-rate (CBR), real-time variable bit-rate (rt-VBR), nonreal-time VBR (nrt-VBR), available bit-rate (ABR), and unspecified bit-rate (UBR). Real-time services would typically receive the highest service priority, nrt-VBR the second priority, and ABR and UBR the lowest priority. Packet scheduling is not a matter for standardization and depends on the switch designer.

5.3. Selective Cell Discarding

Selective cell discarding is based on the cell loss priority indicated by the CLP bit in each cell header. CLP=1 cells should be discarded before CLP=0 in the event of buffer overflows. CLP=1 cells may be generated by a user who deliberately wants to take a risk with excess traffic or might be tagged cells from the UPC mechanism. In a push-out scheme, a CLP=0 cell arriving to a full buffer may be allowed to enter the buffer if a queued CLP=1 cell can be discarded to free space. If more than one CLP=1 cell is queued, the discarding policy can push out CLP=1 cells starting from the head or tail of the queue. Pushing out from the tail of the buffer tends to favor more CLP=1 cells, because the CLP=1 cells left near the head of the buffer are likely to depart successfully from the buffer. If CLP=1 cells are pushed from the head of the buffer, the CLP=1 cells left near the tail of the buffer will take longer to depart and will have a higher risk of being pushed out by the next arriving CLP=0 cell.

More complicated buffer management strategies are possible. For example, a partial buffer sharing strategy can use a threshold; when the queue exceeds the threshold, only CLP=0 cells will be admitted into the buffer and arriving CLP=1 cells will be discarded. This strategy ensures a certain amount of space will always be available for CLP=0 traffic. Similarly, it is possible to impose an upper limit on the number of CLP=1 cells queued at any one time, which would ensure some space to be available for only CLP=0 cells.

5.4. Explicit Forward Congestion Indication

ATM included EFCI as a means for ATM switches to communicate simple congestion information to the user to enable end-to-end control actions. User cells are generated with the second bit in the 3-bit PT field set to 0, signified as EFCI=0. Any congested ATM switch can set EFCI=1 which must be forwarded unchanged to the destination user. The algorithm for deciding when to activate EFCI is chosen by the network provider.

EFCI is used for the binary mode of the ABR service (12). The ABR service is intended to allow rate-adaptable data applications to make use of the unused or "available bandwidth" in the network. An application using an ABR connection is obligated to monitor the receipt of EFCI=1 cells and change its transmission rate according to a predefined rate adaptation algorithm. The objective is to match the transmission rate to the instantaneous available bandwidth. The ATM switch buffers should be designed to absorb the temporarily excessive traffic caused by mismatch between the actual transmission rate and the available bandwidth. In return for compliance to the rate adaptation algorithm, the ATM network should guarantee a low cell loss ratio on the ABR connection (but no guarantees on cell delay).

5.5. Closed-Loop Rate Control

The binary mode of the ABR rate adaptation algorithm involves gradual decrementing or incrementing of an application's transmission rate. The rate adaptation algorithm for the ABR service also allows an optional explicit mode of operation where the ATM switches along an ABR connection may communicate an exact transmission rate to the application. A resource management cell indicated by a PT=6 field is periodically inserted into an ABR connection and makes a complete roundtrip back to the sender. It carries an "explicit rate" field that can be decremented (but not incremented) by any ATM switch

along the ABR connection. The sender is obligated to immediately change its transmission rate to the value of the explicit rate field, or the rate dictated according to the binary mode of rate adaptation, whichever is lower.

6. ATM-Layer OAM

The ATM protocol defines OAM cells to carry out various network management functions in the ATM layer such as fault management and performance management (13). ATM switches are responsible for the generation, processing, forwarding, and termination of OAM cells according to standardized OAM procedures. OAM cells have the same cell header but their payloads contain predefined fields depending on the function of the OAM cell. F4 OAM cells share a virtual path with user cells. F4 OAM cells have the same VPI value as the user cells in the virtual path but are recognized by the pre-assigned virtual channels: VCI=3 for segment OAM cells (relayed along part of a route) or VCI=4 for end-to-end OAM cells (relayed along an entire route). F5 OAM cells share a virtual channel with user cells. F5 OAM cells have the same VPI/VCI values as the user cells in the virtual channel but have these pre-assigned PT values: PT=4 for segment OAM cells and PT=5 for end-to-end OAM cells.

6.1. Fault Management

OAM cells are used for these fault management functions: alarm surveillance, continuity checks, and loopback testing. If a physical layer failure is detected, a virtual connection failure will be reported in the ATM layer with two types of OAM cells: alarm indication signal (AIS) and remote defect indicator (RDI). AIS cells are sent periodically "downstream" or in the same direction as user cells effected by the failure to notify downstream switches of the failure and its location. The last downstream ATM switch will generate RDI cells in the upstream direction to notify the sender of the downstream failure.

The purpose of continuity checking is to confirm that an inactive connection is still alive. If a failure has not been detected and no user cells have appeared on a virtual connection for a certain length of time, the switch on the sender's end of a virtual connection should send a continuity check cell downstream. If the switch on the receiver's end of the virtual connection has not received any cell within a certain time in which a continuity check cell was expected, it will assume that connectivity was lost and will send a RDI cell to the sender.

An OAM loopback cell is for testing the connectivity of a virtual connection on demand. Any switch can generate an OAM loopback cell to another switch designated as the loopback point. The switch at the loopback point is obligated to reverse the direction of the loopback cell to the originator. The failure of a loopback cell to return to its originator will be interpreted as a sign that a fault has occurred on the tested virtual connection.

6.2. Performance Management

OAM performance management cells are used to monitor the performance of virtual connections to detect intermittent or gradual error conditions caused by malfunctions. At the sender's end of a virtual connection, OAM performance monitoring cells are inserted between blocks of user cells. Nominal block sizes may range between 2^7, 2^8, 2^9, or 2^{10} cells but do not have to be exact. The OAM performance monitoring cell includes fields for the monitoring cell sequence number, size of the preceding cell block, number of transmitted user cells, error detection code computed over the cell block, and timestamp to measure cell delay. The switch at the receiver's end of the virtual connection will return the OAM cell in the reverse direction with additional fields to report any detected bit errors and any lost or misinserted cells. The timestamp will reveal the roundtrip cell delay. The measurements of cell loss and cell delay reflect the actual level of QoS for the monitored virtual connection.

7. Bibliography

1. ITU-T Rec. I.361, *B-ISDN ATM-Layer Specification*, Geneva, July 1995.

2. ATM Forum, *ATM User-Network Interface (UNI) Specification Version 4.0*, April 1996.

3. T. Chen, S. Liu, *ATM Switching Systems*, Boston: Artech House, 1995.

4. N. Endo, T. Kozaki, T. Ohuchi, H. Kuwahara, S. Gohara, Shared buffer memory switch for an ATM exchange, *IEEE Trans. Commun.*, 41: 237-245, Jan. 1993.

5. M. Karol, M. Hluchyj, S. Morgan, Input versus output queueing on a space-division switch, *IEEE Trans. Commun.*, 35:1347-1356, Dec. 1987.

6. T-Y. Feng, A survey of interconnection networks, *IEEE Commun. Mag.*, 14: 12-27, Dec. 1981.

7. X. Chen, A survey of multistage interconnection networks in fast packet switches, *Int. J. Digital and Analog Cabled Sys.*, 4:33-59, 1991.

8. J. Hui, Switching integrated broadband services by sort-banyan networks, *Proc. of the IEEE*, 79: 145-154, Feb. 1991.

9. Nojima, Integrated services packet network using bus matrix switch, *IEEE J. Selected Areas in Commun.*, SAC-5: 1284-1292, Oct. 1987.

10. Y. Yeh, M. Hluchyj, A. Acampora, The Knockout switch: a simple, modular architecture for high-performance packet switching, *IEEE J. Selected Areas in Commun.*, SAC-5: 1274-1283, Oct. 1987.

11. ITU-T Rec. I.371, *Traffic Control and Congestion Control in B-ISDN*, Geneva, July 1995.

12. ATM Forum, *Traffic Management Specification Version 4.0*, April 1996.
13. ITU-T Rec. I.610, *B-ISDN Operation and Maintenance Principles and Functions*, Geneva, July 1995.

8. Cross references

9. Reading list

Several good surveys of ATM switch fabric architectures can be found:

H. Ahmadi, W. Denzel, A survey of modern high-performance switching techniques, *IEEE J. Selected Areas in Commun.*, 7: 1091-1103.
A. Pattavina, Nonblocking architectures for ATM switching, *IEEE Commun. Mag*., 31: 38-48, Feb. 1993.
E. Rathgeb, T. Theimer, M. Huber, ATM switches – basic architectures and their performance, *Int. J. Digital and Analog Cabled Sys*., 2: 227-236, 1989.
F. Tobagi, Fast packet switch architectures for broadband integrated services digital networks, *Proc. of the IEEE*, 78: 133-178, Jan. 1990.
R. Awdeh, H. Mouftah, Survey of ATM switch architectures, *Computer Networks and ISDN Sys.*, 27: 1567-1613, 1995.

A wealth of papers can be found on performance analysis of switch architectures, for examples:

A. Pattavina, G. Bruzzi, Analysis of input and output queueing for nonblocking ATM switches, *IEEE/ACM Trans. Networking*, 1: 314-327, June 1993.
D. Del Re, R. Fantacci, Performance evaluation of input and output queueing techniques in ATM switching systems, *IEEE Trans. Commun.*, 41: 1565-1575, Oct. 1993.

The difficulties of constructing large ATM switches are explored in:

T. Banwell, R. Estes, S. Habiby, G. Hayward, T. Helstern, G. Lalk, D. Mahoney, D. Wilson, K. Young, Physical design issues for very large scale ATM switching systems, *IEEE J. Selected Areas in Commun.*, 9: 1227-1238, Oct. 1991.
T. Lee, A modular architecture for very large packet switches, *IEEE Trans. Commun.*, 38: 1097-1106, July 1990.

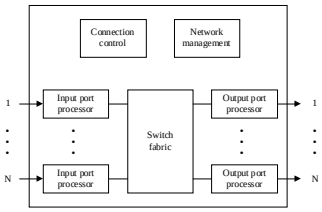

Figure 1. A generic ATM switch architecture

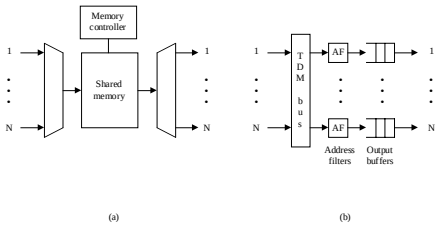

Figure 2. Prototypical switch fabric designs based on (a) shared memory (b) shared medium

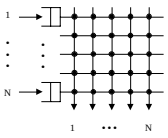

Figure 3. NxN crossbar switch fabric

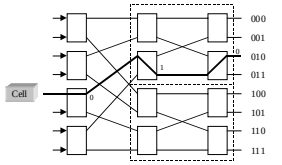

Figure 4. Example of an 8x8 banyan network

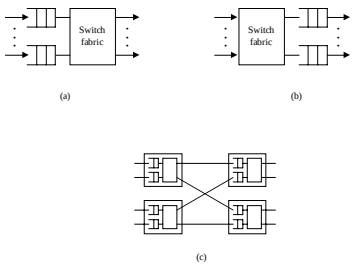

Figure 5. Examples of (a) input buffering (b) output buffering (c) internal buffering

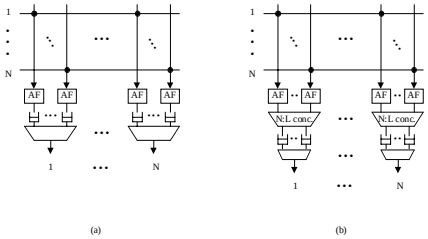

Figure 6. (a) Crossbar switch with buffers at each of N^2 crosspoints (b) Knockout switch with NL buffers

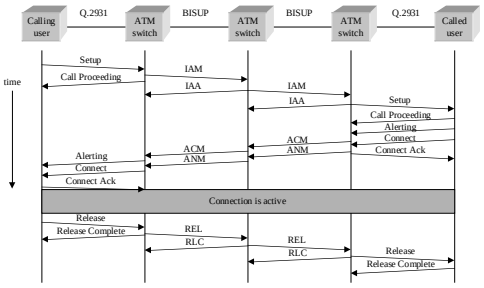

Figure 7. Exchange of signaling messages involved in a successful connection

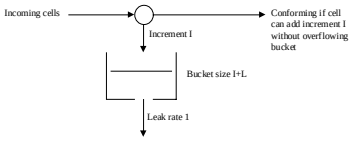

Figure 8. GCRA operation viewed as a leaky bucket algorithm