

Seuraavilla sivuilla käsitellään ATM-verkkojen liikenteen hallintaa eri näkökulmista. Ensimmäiset kolme lukua muodostavat yleisen mutta suppean johdannon ATM-tekniikkaan. Luvussa 4 pyritty hahmottamaan ATM-verkon liikenteen hallinnalle asetettavia vaatimuksia. Luvussa 5 on esitetty ATM-verkon liikenteen hallinnan peruskomponentit ja viimeisessä luvussa on oma käsitykseni, mutta ei mikään ehdoton totuus, siitä miten alkuvaiheen ATM-verkon liikenteen hallinta tulisi toteuttaa. Lähteinä olen käyttänyt väitöskirjaani, kahta teletekniikan laboratoriossa kirjoitettua seminaariesitelmää sekä yhtä julkaisematonta suomenkielistä raporttia.

1. ATM principles

1.1 Background

The first article about the principles of ATM appeared eleven years ago (Coudreuse 1983). According to Coudreuse the basic target of ATM was to meet the challenge that changing telecommunications needs posed to the techniques and technologies of information transport. Although at present ATM is almost unanimously accepted as the basis for broadband networks, there have been other alternatives. The two basic alternatives are packet networks and digital networks based on the synchronous transfer mode. Let us discuss briefly the weaknesses of these alternatives in order to clarify the strength of ATM in an integrated services environment.

In a typical packet network one connection reserves the whole link during the transmission of a packet. As a result all other connections must wait until the transmission is ended. This is a substantial disadvantage in a multiservice environment because some applications, especially voice, are sensitive to delay. Although priorities can be used to alleviate this problem, it is usually not possible to break off the transmission of a packet in order to send more urgent packets.

Synchronous digital networks, particularly narrow-band ISDN (Integrated Services Digital Network), offer another alternative to ATM. In the ISDN the basic unit is 64 kbit/s channel, which is without doubt suitable for vocal communications. However, the somewhat inflexible structure of ISDN has a number of disadvantages as regards other services. The channel is reserved all the time irrespective of the actual capacity needs of the transmission. In addition to 64 kbit/s ISDN sustains only a few other rates such as 2 Mbit/s (or 1.5 Mbit/s). If a service uses more than one 64 kbit/s channel, different channels are routed through the network independently and it is difficult to guarantee that all channels have an equal delay.

The fundamental difference between ISDN and ATM is that instead of fixed speed ATM network uses fixed packets, called cells. The size of an ATM cell is 53 bytes (424 bits) of which 5 bytes are used for header and 48 for user information. Although the basic unit in ATM technology is the amount of information, there are standardised bit rates for ATM interfaces, namely, 155 and 622 Mbit/s. These bit rates and the cell size determine the time units of ATM networks, 2.73 μ s and 0.68 μ s, which are the transmission times of one cell at 155 Mbit/s and at 622 Mbit/s, respectively. Nevertheless, it should be noted that ATM principle can be adapted for any other bit rate.

Flexibility was given priority at an early stage of ATM development partly because of inability to predict service demand. Flexibility is achieved by an intrinsic property of ATM: all types of information (voice, data, video and still picture) are presented in the same form using equal-sized cells. There has been, on the other hand, considerable suspicion as to the viability of the integration of services with very different characteristics: Voice and video are intrinsically analogue signals; A

typical data source produces variable length packets according to almost an unpredictable process; Still pictures, such as X-ray pictures, may have a huge amount of information that should be delivered through the network in a couple of seconds.

The application of a relatively small information unit results in excessive segmentations and re-assemblies, especially for data flows, and consequently additional headers and computation and possibly an increased probability of information loss. These are the main disadvantages of combining in one network voice and video services with data services. Despite these problems, the integration principle seems to be better than using separate networks for different applications, and ATM seems to be the most efficient technique to combine various traffic flows into the same network (Figure 1.1).

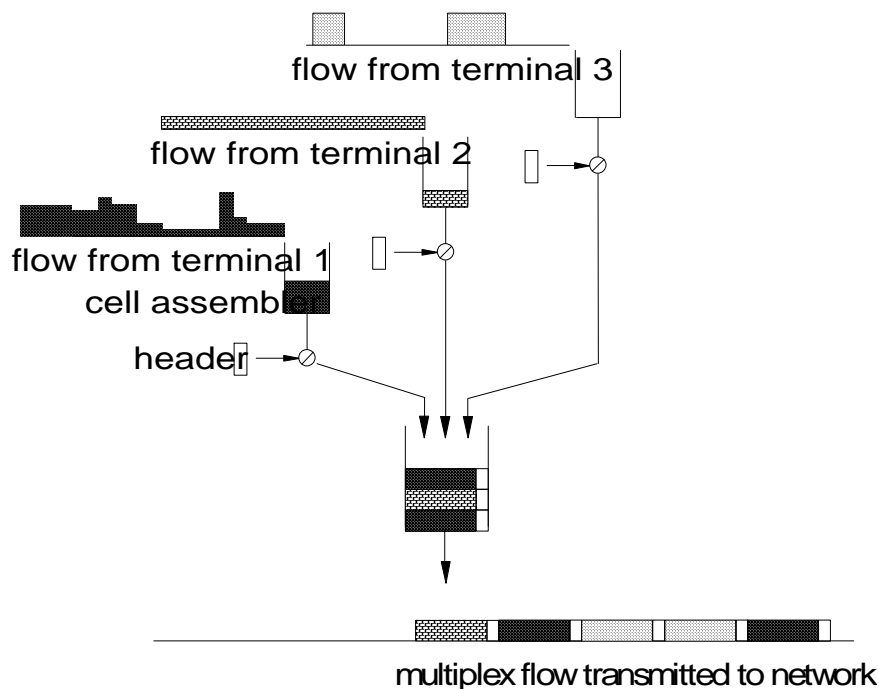


Figure 1.1. The multiplex principle of ATM networks.

In ATM networks both segmentation and buffering have the ability to alter the properties of the traffic stream. If the original traffic stream has any correlation with previous events or with other connections, the segmentation in ATM interface does not change or changes only slightly the characteristics of correlation. If all data packets are split into cells and then sent to the network one after another, the buffer requirement inside the ATM network is in principle the same as in a pure data network without segmentation. Nevertheless, there is a considerable change because the segmentation offers the *opportunity* to alter traffic flows more precisely. This is perhaps the main advantage of the ATM network as compared with packet networks. Although the ATM cell is a kind of packet, the cell size is so small that the additional delay due to transmission of one cell is negligible and even the emptying time of a full buffer is usually short in comparison with the propagation delay.

The effect of buffering depends on the buffer capacity. In this study the basic assumption is that the actual buffer capacity in ATM networks is relatively small, typically 100 cells, and if bigger buffers are necessary, they are situated outside the core network. As regards traffic analysis, the alteration effect of large buffers can be taken into account in the incoming traffic process. Because of the small buffers, the delay at network nodes does not have a substantial effect on most applications and, in consequence, the cell loss probability is usually the factor that sets a limit on the network utilisation.

1.2 Introduction to ATM traffic modelling

The overall traffic process in ATM networks will be extremely complicated. One way to facilitate the analysing of ATM traffic is to divide the traffic process into several levels each of which has its typical time scale and typical traffic characteristics. In this study a resolution of four time scales is applied:

- In *call scale* traffic variations are caused by the call process. These variations are usually managed by means of Connection Admission Control.
- *Rate-variation scale* includes the variations induced by the changes in required cell rate, for example in a Variable Bit Rate video or audio connection. This scale covers typically the region from 20 ms to minutes.
- In *burst scale* the inherent phenomenon is the arrival process of bursts, such as arrivals of packets from Local Area Networks. This time scale covers typically the region from 0.1 ms to 100 ms.
- In *cell scale* each connection can be supposed to be deterministic (the interarrival time between successive cells is constant) and thus the variations in aggregated traffic process are due to randomness of phases of different connections. The time scale of these variations is approximately from 1 μ s to 1 ms.

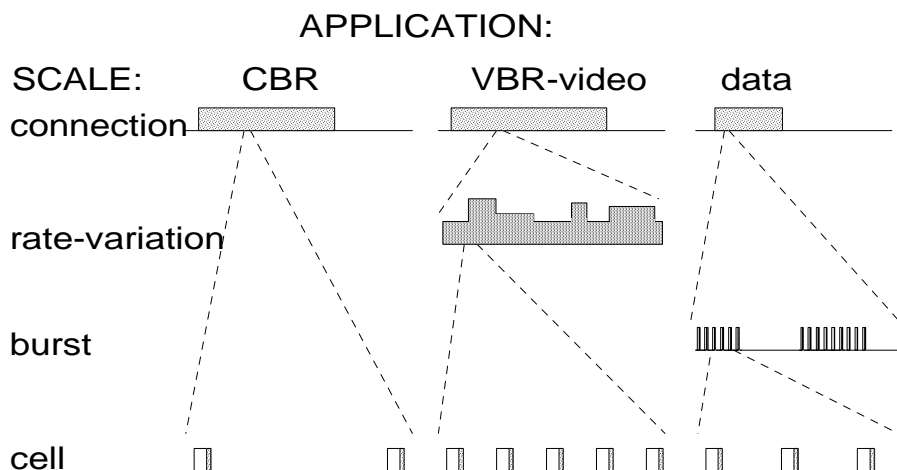


Figure 1.2. Time resolution of ATM traffic process.

A time resolution with three, four or even more scales has been applied widely. There are no problems to determine and name the connection and cell scales whereas the situation is much more difficult with the intermediate scales. The names used in this study try to depict the inherent characteristics of traffic process at each scale.

A burst is interpreted as a block of information which has a certain size but not necessary a tight requirement for the peak rate during bursts (except that the transmission of a burst should end before the arrival of the next burst). In contrast, in rate-variation scale there is typically no definite amount of information but a required level for the average cell rate. In this scheme all traffic processes in burst and rate-variation scales are covered by one (burst) scale. However, this unification of all traffic processes between cell and call scales is unsuitable for use in this study.

The traffic analysis in this study covers mainly the three lowest scales (cell, burst and rate-variation) but not the variations induced by the call process. The call process is the prime phenomenon as regards the network dimensioning, and in some cases it may be very difficult to distinguish the call scale process from the processes of other scales. A typical situation is when a Virtual Path (VP)

cross connect network is used to transmit LAN traffic. Virtual Paths are typically permanent and the traffic tends to vary considerably during the holding time of a VP connection because virtual connections are established and released. As the virtual connections are transparent for a VP network, it is possible that a CAC procedure has to deal with traffic variations of a relatively long time scale. In traffic analysis these variations can be included in rate-variation scale models.

The aggregated process including cell, burst and rate-variation scale fluctuations is extremely difficult to analyse mathematically. However, a regular behaviour can be found independent of the actual traffic parameters in each scale (see Figure 1.3). Several studies have shown that, when VBR connections are aggregated on a multiplexer, the queue length distribution is composed of two distinct components. In this study a model with three components has been applied. The additional rate-variation scale component (horizontal line in Figure 1.3) arises where the input rate is permanently greater than the output rate. The burst scale component (the middle component in Figure 1.3) is due to relatively short bursts which can be partly buffered even by the small buffers of ATM nodes. The cell scale component (the steepest line in Figure 1.3) reflects the small queues which occur due to the asynchronous arrival of cells from distinct connections.

The queue components are not necessarily as clear with real traffic processes. There are regions in which the two scales are effective at the same time (rounded edges). Furthermore, since the overload periods are infinitely long only in theoretical models not in practical situations, the rate-variation scale component is horizontal only in theory and the boundary between the burst and rate-variation scales may entirely vanish.

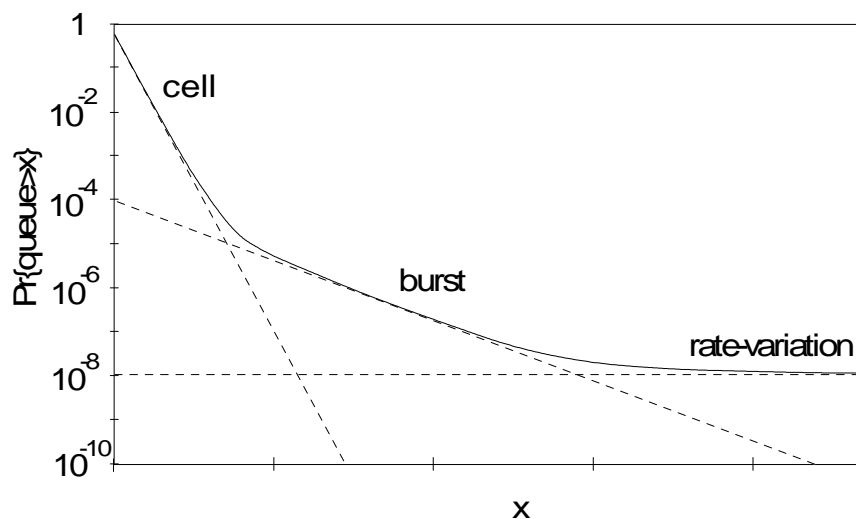


Figure 1.3. Cell, burst and rate-variation scale components of queue length distribution.

2. Current situation of ATM technology

Let us first consider the main advantages of ATM technology. There must be solid ground for introducing a completely new technology in the existing telecommunication environment. Without this ground the future of the new technology will be, usually after a short phase of enthusiasm, very uncertain. It is easy to find several examples of this kind of life cycle in telecommunications: ISDN (Integrated Services Digital Network) as the most famous case, and DQDB (Distributed Queue Dual Bus) seems to be one of the recent downfalls.

So a vital question is if there is a real demand which is very difficult to satisfy with any other technology. Our opinion is that there really is although we have to be cautious when predicting the customer demand. The present condition in the telecommunication and information technology gives us some valuable indications. Firstly, we cannot pass over the Internet boom, and the

problems entailed. These problems, especially the poor Quality of Service (QoS) of Internet, certainly need good solutions. Secondly, new video services are steadily emerging, especially if there were enough capacity in all parts of the network to transmit the bandwidth needed by video services with an adequate QoS. Thirdly, the huge investments needed for the realization of a (wide area) broadband network imply that there must be a workable billing system with an appropriate tariff structure for different services. All these issues should be thoroughly considered independently of the technology selected for the implementation.

As a consequence, any broadband technology intended to public networks must have the following properties:

- scalability from some megabits per second to tens of gigabits per second,
- suitability for integrated real-time services with video and audio components, and
- a proper network management system including billing.

It seems to be almost a unanimous opinion that ATM is the most suitable, currently available technology to meet *all* these requirements. But why has ATM not yet been a real success? Let us make a short review on popular answers. J. McQuillan has presented 10 reasons, most of which are connected with the customer or the market. Despite the technological excellence of ATM there are many obstacles before the benefits can be brought to the end-users. The primary obstacles are obviously related more to the software and protocols than to the ATM hardware: the ATM protocol infrastructure with about 200 000 lines of new code may prove to be too complex. It will take years before all the protocol software components are acting together without problems. Even though it might be possible to build the network and offer a service like videoconferencing, the hidden costs of installation and management can be too high in order to justify the introduction of the new service.

One prominent experience of ATM technology is continuously evolving standards. First adopters have faced a dilemma of never ending software and hardware updates. Nowadays the hardware has reached its maturity but the software seems to be updating in steady speed - new software updates are coming out almost monthly. To operate a multivendor environment in such an unstable situation may drive the management people on the verge of despair.

A typical problem is connected with the fact that new applications often require native functions of ATM. Native functionality is brought by Application Programming Interface (API). Unfortunately, API is not yet standardized and therefore constantly changing. In this case the interoperability cannot be guaranteed even in updates of single vendor. The worst case is that you have a switch from one vendor, Network Interface Cards (NIC) from another and an application from a third. When a new UNI specification is finished vendors race for a first update. On that race updates come out on the time scale of half year. If you update your switch you usually lose all the connectivity to the old version of NICs. If you update your NICs you lose your connectivity to the switch and application (which is presumed to use API and which has changed as well). This lack of interoperability forces to wait until all of the vendors have finished their updates, after that you may step one step further and update your whole network. This total update would require time for equipment to be out of service and therefore it might be quite a harmful event.

From the operating point of view there seems to be only little problems as long as someone is willing to do the hard work of connection configurations. Signaled connections are easy to implement in small scale but large networks still need a lot of handwork. Setting up connections is relatively easy when you have one protocol to do that. In contrast, if you have more than one protocol in your hand - as it is with ATM, the situation is much more complicated. Standardized

signaling protocols have been developing at a steady speed of one release per year. Interoperability of new releases has been poor and, moreover, the lack of sophisticated functionality in standardized protocols has impelled some vendors to create their own signaling protocols which are not so promising for multivendor environment.

Nevertheless, the market leader Fore has managed to conquer a big share of the market with their SPANS protocol. SPANS has features which are needed when more than one switch operates in an ATM network. This evolution of protocols where several parties are competing with each other, will be prevalent until the standardization process is completed (if that will ever happen). Network operators and users can only attempt to adapt to the situation. In consequence, many potential users of ATM may suspend their investments in ATM technology due to the uncertain future of current equipment.

2.1 Services

Network is worth of nothing if it has no services. Services are implemented through applications which are steadily but slowly moving towards native ATM. When talking about using applications in an ATM network we cannot yet expect them all to be plug-and-play. In this respect the applications supposed to use ATM network are still imperfect. Actually, in order to really make a breakthrough of ATM technology we must develop ATM applications which are acceptable also by others than computer-wizards.

As we explained earlier, one of the primary limitations is a frequent updating of the software and the hardware relating with ATM arising from constantly evolving standards. Compatibility problems are due to this fact. In order not to run to these problems one has to be aware of evolution process: for example what is the latest release of an ATM-driver in a workstation and what are the applications supporting it.

Another problem area is the applications using Proxy Signaling and Permanent Virtual Circuits. This means that a person who wants to use these applications has to change configurations in an ATM switch. To do that might not be an arduous task, but it certainly will be an obstacle or even a stumbling block to a person who is not used to doing these things - most people are not very familiar with the delicate characteristics of ATM.

Another key service, which is said to solve all problems, is LAN Emulation (LANE). LAN Emulation has been developed to assist easy adoption of ATM. LAN Emulation should emulate characteristics of legacy networks in a way that all of the legacy LAN programs and services would run on the top of ATM. LAN Emulation is simply conversion between legacy addressing and ATM addressing. In the case of Ethernet emulation we have MAC to ATM conversion. What could be so difficult with this? We have had bridges, routers and gateways which have done the same thing. Yes, this is true but in the case of LAN Emulation we have to emulate a connectionless network on top of a connection oriented network. This arrangement results in a tremendous signaling load to the network. In order to cope with this load LANE has been divided into three parts: a client and two servers. A client is a process residing in the workstation or PC which communicates with servers to find an appropriate ATM address to a certain MAC address. Servers are responsible for information gathering. They do all the raw work on the network to find out right conversion for addresses. To configure these functions properly in the case of more than one switch appears to be a work that needs a real expert or people who are willing to spend their nights beside switches.

2.2 Experiences

To study and exploit new ATM services Laboratory of Telecommunication Technology has driven a project to establish a test platform for new services. This platform has been built up in close cooperation with VTT and Laboratory of Communications. Our laboratory has 2.5Gbps switch (Fore ASX-200) which is equipped with 12 155Mbit/s OC-3c optical interfaces and 4 155Mbit/s UTP-5 twisted-pair interfaces. To have desktop connectivity we have two switches (Madge Collage 280) with 25Mbit/s twisted-pair interfaces. To be able to exploit different services we have acquired a video compressor Nemesys AVA-300 and a decompressor Nemesys ATV-300, network interface cards (Fore SBA-200, Fore PCA-200, Madge ATM 25) and a measurement tool (Adtech AX/4000). The environment is presented in Fig. 2.1.

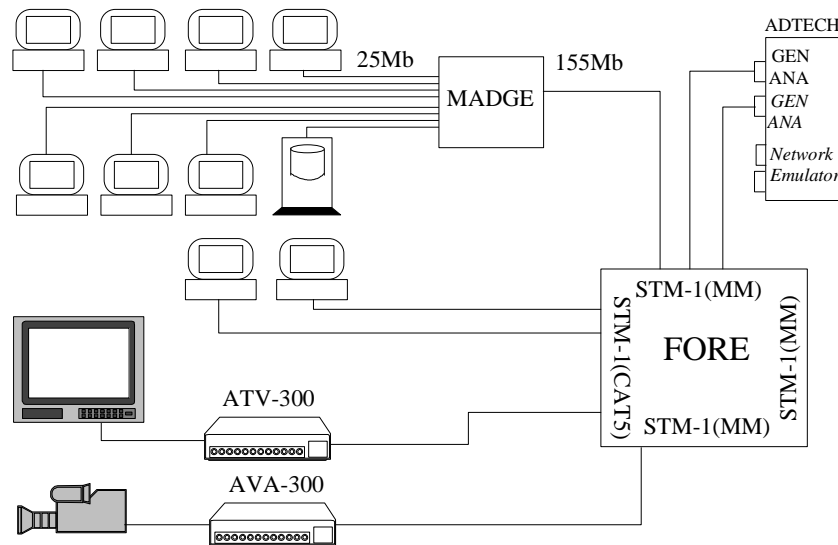


Figure 2.1. Laboratory environment

Good examples of complexity of current ATM environment are Nemesys AVA-300 and ATV-300. AVA-300 is a device which converts analogue video signal into a digital format, either raw or M-JPEG-compressed, and encapsulates it to ATM cells using AAL5 frames. ATV-300 decompresses M-JPEG-video streams and makes the digital-to-analogue conversion. AVA can work both in PVC mode or using multicast Switched Virtual Circuits but in the latter case it needs manager process, which handles signaling for the device. In order to implement this kind of proxy signaling one still needs to create a couple of PVC, the amount depending on the signaling protocol used.

Software updates have proved to be difficult. New software for the switch comes out every half a year and for NICs little bit more often. It has been common that API has changed with every update on NICs or even if it has not changed, there has been safety lock which prohibits older application to run over a new API. This has been rather annoying since it takes time from application vendors to change their code. Last problem came out when Fore moved to software release 4 and changed their API. Our Nemesys video converters use API stack for proxy signaling. This change in API was not documented and it took several weeks to find out where problem was.

Another curious problem has been signaling stacks. Although vendors say that their signaling uses ITU-T standardized protocol they do not specify whether it is Q.93B, Q.2931. Actually Q.93B and Q.2931 are the same document but they use different Service Specific Coordination Function (SSCF) and Service Specific Connection Oriented Protocol (SSCOP). UNI 3.0 is based on Q.93B

and UNI 3.1 on Q.2931. If a vendor has come up with some own idea (like some have) there is an imminent conflict between two switches. They seem to recognize others but do not work correctly.

Operation of Madge and Fore switches has been mainly stable and satisfying. They have implemented LAN Emulation and connection management services inside switches. This on-box solution gives a feasible picture of their operability. Initial configuring is quite easy to accomplish as long as the network does not require usage of centralized LAN Emulation. Madge has rather alluring functionality to switch ethernet and ATM traffic. This bridging function allows easy migration to ATM.

We have accomplished some performance studies with the Fore switch. The main goals of the evaluation have been to collect information on the performance measurements in ATM switching systems, and to verify their usability in a real testing environment. The results of the measurements met mainly with the requirements stated to an appropriate ATM switch. The switch performed traffic management functions quite well and consistently although there were some minor deficiencies. This enables the spread of ATM networks, because problems in traffic management have slowed down the widespread utilization of broadband networks. Now when problems with late collisions in Ethernet traffic have been solved, the ATM-network build in our laboratory seems to be a suitable solution for campus environment where budget is small and migration time could be quite long.

2.3 Prospects

In spite of the problems, it is very probable that Internet will be upgraded to ATM in one way or other because of increasing capacity and quality requirements. In contrast, it is not clear what the definitive role of ATM in the whole network architecture will be. One possibility is to exploit the current technology as widely as possible and to use ATM only as a tool that makes possible to meet the three requirements mentioned above. The Internet protocol (IP) switching is a typical example of this approach. The strength of this concept is that it uses ATM switching for speed, but addressing, routing and bandwidth allocation are accomplished via Internet protocols which are supposed to be less complicated. The real usefulness of this approach is still uncertain because there are fundamental management problems in all networks with both real-time and non-real time services.

Another possibility is to aim straightly at a pure ATM network similar to that in our laboratory. This approach requires, in addition to complete ATM network protocols, economical interfaces for workstations and personal computers, enhanced ATM APIs and general acceptance for this scheme among software developers. Our objective has been to investigate the usefulness of a network architecture which makes possible ATM connections from desktop to desktop. Because this network architecture is especially suitable for video applications, we have begun studies relating to the performance requirements of video services in ATM environment. According to our experiment, end-to-end ATM connections are technologically possible, even though it is somewhat difficult to recognize any commercial video application which positively requires a pure ATM connection.

ATM network may as well be applied for more efficient voice transmission because of its intrinsic on/off characteristic. This approach seems to have even now demand among some network operators, and with multiplying data transmission ATM will have an increasing importance in mobile networks. A standardization effort is going on in this area. Unfortunately, completing the standardization may take so long that several proprietary solutions arise making the wide employment of voice over ATM troublesome.

As a consequence of these prospects there is an impending danger of designing an extremely complicated ATM network with specific functions for all services with several network services having different priorities. Although this strategy may appear to be attractive since it promises lots of excellent properties, it will be very difficult to build the network and manage the services. One approach is to keep the network as simple as possible taking into account only the most significant service demands. This means that in the first phase the network may offer only two or three relatively simple services with which the operator may satisfy the three basic requirements: scalability, service integration and network management.

As a conclusion of experiences it can be said that software updates form the main bottleneck of ATM. It is probable that real breakthrough can be expected when software stabilizes and the API stack is standardized. After that we may suppose that a lot of new and fascinating multimedia applications will emerge.

What could be the conclusion from these insights as regards the teletraffic research? ATM nodes must support both real-time and non-real-time services at the same time, be appropriate to both pure ATM connections and IP-based connections, have an enhanced billing system, and be inexpensive. Every topic has something to do with traffic issues: the coexistence of real-time and non-real-time services induce several fundamental difficulties; the cooperation between IP and ATM worlds is still an unresolved puzzle; and, particularly, tariffs and billing need a lot of theoretical studies and practical experiments. Finally, we must keep the whole system as simple as possible in order to enable inexpensive services.

3. Service types in ATM networks

This section describes the basic properties of various service types in broadband networks. Services can be classified into five main groups: circuit emulation, voice, video, data and multimedia. Each service type has its inherent requirements for ATM networks.

3.1 Circuit emulation

The basic idea of circuit emulation is to hide the ATM nodes and links so that the flexible ATM technology can be brought inside the present telecommunication infrastructure with as few changes as possible. A typical situation is an operator who wants to offer switched $N \times 64$ kbit/s connections for business customers. In the present telephone network, managing these connections is a difficult task whereas in ATM networks the operator can control connections flexibly and offer transmission capacity immediately by means of ATM crossconnects. From the ATM network point of view, circuit emulation connection is a Constant Bit Rate (CBR) source with strict requirement for cell delay variation.

According to many authors a VBR connection should be interpreted as a CBR source determined by the peak rate if the ratio of peak rate to link rate is greater than $1/15$ or $1/20$. Therefore the actual bit rate from a source determined as a CBR connection is not necessarily always the same as the declared peak rate.

3.2 Voice

Although voice communication is frequently considered an insignificant service for broadband networks, it should not be totally ignored. A typical telephone conversation generates more than 10 Mbit information in both directions and, for example, the amount of information transferred by the

Finnish long distance telephone network is roughly 10 Gbit per year and inhabitant. Some data applications, such as remote use of supercomputers, can generate perhaps 1000 times as much information during a year but, on the other hand, applications of this type will only be exploited by a few specialists.

In fact, voice communication has been taken into account in the standardisation of ATM. The cell size is partly determined by the requirements of voice connections because the larger cell size, the longer it takes to gather up a whole cell from the bit stream. This delay is 6 ms for a 64 kbit/s connection and if there are several ATM parts in the path of the connection, these delays, together with the propagation delay, may have a disturbing effect on a telephone connection. For the same reason, large buffers at network nodes are not recommendable. On the other hand, voice connection is usually less sensitive to cell losses than video or data applications.

There is much knowledge of the general behaviour of traffic in telephone networks. However, the situation in ATM network differs from that of the ordinary telephone network since it is possible to adapt to the varying bit rate demands during conversation. Nowadays a telephone call uses a constant 64 kbit/s channel, but this is not what is really needed since both talkers are seldom talking at the same time and, in addition, there are clear pauses between successive words and sentences. Depending on how accurately the silence periods are detected, the proportion of active periods varies from 0.35 to 0.5 (Brady 1969). If we take into account that the 64 kbit/s PCM (Pulse Code Modulation) coding can be replaced by the 32 kbit/s ADPCM (Adaptive Differential PCM) coding without deterioration in speech quality, the average needed bandwidth of a voice connection can be reduced to 12 kbit/s in ATM networks. Consequently, an ATM link with a capacity of 622 Mbit/s may transfer roughly 40 000 telephone calls simultaneously.

3.3 Video

In the long term, the most important type of service of broadband networks is presumably video communications. Video communications consist of a wide variety of services from slow rate videophones to High Definition Television (HDTV) and the required bit rate may vary from tens of kilobits to hundreds of megabits per second. In order to utilise network resources efficiently layered coding schemes have been suggested. The idea of layered coding is, according to Ramamurthy and Sengupta (1990): Applications like broadcast video that require large bandwidth, may use layered coding and mark packets as essential and enhancement packets. Essential packets help to reproduce the basic picture at the receiver and keep the session intact, and hence have to be carried without loss. Enhancement packets enhance the quality of the picture, and can be dropped in the event of congestion in the network without disrupting the session.

Typical properties of video sources with VBR coding are:

- a sharp peak occurs when the scene changes but variations are relatively slight for the same scene;
- the form of stationary distribution depends on the type of sequence (videophone, videoconference, entertainment) and on the coding method;
- the autocorrelation function decreases rapidly over the first few frames but the rate of decrease then slows down;
- if burst scale traffic variations are buffered the necessary buffer capacity might become very large;
- if complicated coding methods are used, cell loss rates for the important data shall be very low, in the order of 10^{-10} , whereas it may be possible to tolerate a greater cell loss rate for the remainder;

- video phones and video conferences will require all their packets to be delivered without delay;
- a video connection is seldom used without voice and other service components; this can bring about correlation between different connections and complicate the traffic control.

Kawashima and Saito (1990) have presented a summary of video source models with three bit rate parameters: mean (m), standard deviation (σ) and maximum (h). Since these models are concerned with a wide range of sources from videophones to studio television, it is not reasonable to take direct averages from these figures, instead we can use parameters such as ratios of standard deviation to mean and mean to peak. From Table 3 in (Kawashima & Saito) we can obtain the following average values for these parameters:

$$\frac{1}{M} \sum_{i=1}^M \frac{\sigma_i}{m_i} = 0.36,$$

$$\frac{1}{M} \sum_{i=1}^M \frac{m_i}{h_i} = 0.41.$$

These figures can be obtained by the following bit rate (λ) distribution (the mean bit rate is 1 Mbit/s):

$$\Pr\{\lambda = 0.86 \text{ Mbit/s}\} = 0.60,$$

$$\Pr\{\lambda = 1.00 \text{ Mbit/s}\} = 0.34,$$

$$\Pr\{\lambda = 2.41 \text{ Mbit/s}\} = 0.06.$$

Although the real bit rate distribution may be much more complicated, these figures contain the essence of the video source: there is relatively stable behaviour most of the time (0.88 and 1 Mbit/s in the example) and intermittent periods with a substantially higher bit rate requirement (2.41 Mbit/s).

3.4 Data

One definition for data communication is that it consists of all possible applications which use a computer as terminal equipment, in fact, everything that is not voice or video can be classified as data. Up to now it has been possible to distinguish data applications from voice and video, but the recent development of telecommunication services has blurred the edges between different service types. Let us take for example a video art library. A fraction of a video movie has been saved on a computer disk and then manipulated by means of a sophisticated program that changes essentially the original content of the video. Then the edited video is sent automatically to the network after a request from a customer who uses it as a part of a multimedia application. It is not at all clear whether the result is data, video or some other type of connection.

In this study we are dealing with ATM traffic and its characteristics (there are many other viewpoints but they are not considered here). Consequently, the primary issue is what requirement an application or user has, in particular, whether or not the bandwidth requirement at any given time is determined by the source. The prime issue in the previous example is therefore whether the video tape is played immediately or saved on disk and played afterwards. In the latter case the used bit rate during transmission may be low or high depending on the charging policy and network load at the time, and it may vary independently of the actual content of the original source. A connection with these properties should be classified as a data connection rather than a video one.

Similarly, the properties of a data connection may approach those of video. For example, if a designer utilises computer aided animation remotely by aid of a supercomputer, user requirements are similar to a typical video connection even though no real video camera has been used. Thus the demands an application makes on the network are more important than the type of terminal. This must be taken into account in the source description: it is not possible to classify all sources to pre-defined groups according to the terminal type and other permanent information because the user application is in many cases more important than the terminal type.

In consequence, there is no typical data connection and no single feature appropriate to every data connection but, nevertheless, there are some typical characteristics for most data traffic:

- the bit rate needed can be very high but usually the peak rate is used only over a small fraction of time and thus the mean rate is much lower than peak rate;
- long bursts of information are interspersed with short messages (e.g., acknowledgements);
- unknown and unpredictable on- and off-period statistics;
- loss tolerance depends on the coding scheme;
- sources are controllable in the sense that they can be slowed down, if needed, without affecting the viability of service.

There are two fundamentally different types of situation with respect to the traffic control in ATM networks:

- *individual connections*, which means that the original properties of connections are visible to the ATM network;
- *aggregated process*, typically between Local Area Networks, when the ATM network have little, if any, knowledge of the actual connections used by different applications.

According to Doshi et al. (1991) the former group can be divided into three different data types:

- *Relatively smooth data* comes from sources for which the cell arrival process is not as periodic as for CBR sources but the ratio of mean rate to peak rate is relatively high, say ≥ 0.1 , and the bursts at peak rate are relatively short and nearly constant.
- *Bursty interactive traffic* and short intermittent file transfer are characterised by a relatively small value of the mean rate to peak rate requirement (could be < 0.01), and data bursts at peak rate ranging from a few bytes to a few hundreds of kilobytes.
- *Bursty long file transfers* correspond to long infrequent file transfer. Typically, the idle periods between such file transfer are much longer than the time to transmit file, implying a small ratio of mean to peak bandwidth requirement. These sources are delay tolerant.

The packet length distribution depends on the application but typically it has clear peaks at the minimum and maximum packet sizes. According to Falaki and Sørensen (1992) the best fit to the interarrival time distribution on a local area computer network is provided by a hyperexponential distribution with two contributing terms: 68% of the traffic has a mean interarrival interval of 25.2 ms and the remaining 32% has a significantly larger mean interarrival interval of 235.2 ms.

Data connection models may be complicated but much more difficult is to determine general LAN interconnection traffic because essential information (what the main properties of connections are, when they start and end and so forth) is either uncertain or unknown..

3.5 Multimedia

A multimedia call may consist of audio, video and data components, and traffic control can treat these components as separate connections. However, problems may arise because of the interdependence of separate connections inside a multimedia call, for example between audio and video components. Unfortunately, almost every model for the aggregate traffic process relies on the assumption that different connections are independent of each other. This intricate problem needs further study but because we do not have enough knowledge of real multimedia traffic and its interdependencies.

4. ATM-verkkojen liikenteen hallintaan

4.1 Johdanto

Tämä selostus pyrkii muodostamaan yleiskuvan ATM-verkon liikenteen hallintaan liittyvistä kysymyksistä, jotta erilaiset käsitteet, ongelmat ja ratkaisumallit voitaisiin sijoittaa järkevään kokonaisuuteen. Muistetaan kuitenkin, että syvällisen asiantuntemuksen kehittäminen vie aikaa (siis vuosia), joten yhden esityksen puitteissa ei voida edetä kuin hieman.

Heti alussa on kiinnitettävä huomiota siihen, minkälaisia malleja asioiden (tässä tapauksessa siis laajakaistaverkon toiminnan) esittämiseen käytetään eri yhteyksissä. Asiaan perehtyneillä henkilöillä on välttämättä jonkin tyyppinen kokonaiskuva valmiina; ongelmaksi voi muodostua eri henkilöiden kuvien keskinäiset erot, johtuen esimerkiksi henkilöiden erilaisesta koulutustaustasta. Toisaalta asiaa tuntemattomalla henkilöllä kuvaa ei ole lainkaan tai se on hyvin hämärä, jolloin kommunikointi voi muodostua vaikeaksi ellei tarjolla ole valmiiksi mietittyä, hyvin jäsennettyä mallia.

Mallit voidaan jakaa karkeasti kolmeen tasoon. Verkon hallinnasta vastaavalla, eli operoivalla henkilöllä mallin (M_o) tulee olla huomattavasti yksityiskohtaisempi kuin tavallisella käyttäjällä (M_u), mutta sen ei kuitenkaan tarvitse olla yhtä täydellinen kuin järjestelmän kehittäjällä (M_d, joskin yksityiskohtainen ja samalla kokonaisvaltainen näkemys on yleensä vain muutamalla henkilöllä). Operoijan malli on erittäin keskeinen verkon hallinnan kannalta, kun taas käyttäjän malli vaikuttaa olennaisesti siihen miten käyttäjät arvostavat operaattoriltaan saamaansa palvelua.

Käyttäjien verkkomallilla voi siten olla vaikutus myös siihen miten asiakas kokee palvelun laadun. Ongelmatilanteissa saattaa olla hyvinkin hyödyllistä, että käyttäjä ymmärtää jossain määrin verkon sisäistä toimintaa, jolloin esiintyvät vaikeudetkin voivat olla jollain tasolla ymmärrettäviä ja hyväksyttäviä. Tällaisesta tilanteesta voidaan ottaa esimerkiksi Internetin puhelinpalvelu, joka poikkeaa puhelinverkon muodostamasta standardipalvelusta johtuen verkon ja sen protokollien erityisominaisuuksista.

Huomattakoon, että tässä yhteydessä pyritään kehittämään järjestelmämallia (M_d) pitäen mielessä sekä käytännön toteutettavuus että mallin yksinkertaistukset (M_o ja M_u). Itse asiassa pyrkimyksenä on rakentaa esitys siten, että ensin rakennetaan melko karkea käyttäjätason malli, jota sitten täydennetään siten, että se riittää verkon operaattorille ainakin rutiinitehtävissä. Täydellisimmän järjestelmämallin tulisi olla niin yksityiskohtainen, että siihen ei ole mahdollista päästä tämän esityksen puitteissa. Tällä tavoin rakennettuna on ehkä kuitenkin mahdollista pitää eri tason mallit rakenteellisesti yhdenmukaisina, siten että mallista toiseen on mahdollista siirtyä sujuvasti.

Millä tavoin malleja tulisi sitten rakentaa. Ensiksikin tulee luoda mahdollisimman yksinkertainen kuvaus verkon rakenteesta ja sen osista. Eräs varteenotteva mahdollisuus on kuvata verkon osien toiminta objekteina, joilla on tiettyjä tehtäviä tai tavoitteita. Näitä varten objekti tarvitsee ominaisuuksia, joita pyritään sitten kehittämään parhaalla mahdollisella tavalla. Kokonaisuuden hallitsemiseksi tarvitaan erilaisia työkaluja, joihin voi sisältyä myös simulointimalli tai jossain tapauksissa myös analyttinen malli. Jos mahdollista, tällaisen työkalun tulisi olla kiinteä osa kirjallista selostusta (tässä voisi ajatella esimerkiksi simcity-tyyppistä ohjelmaa, joskaan sellainen ei kuulu tämän esityksen tavoitteisiin).

4.2 Atm-verkko ja sen liikenteen hallinta

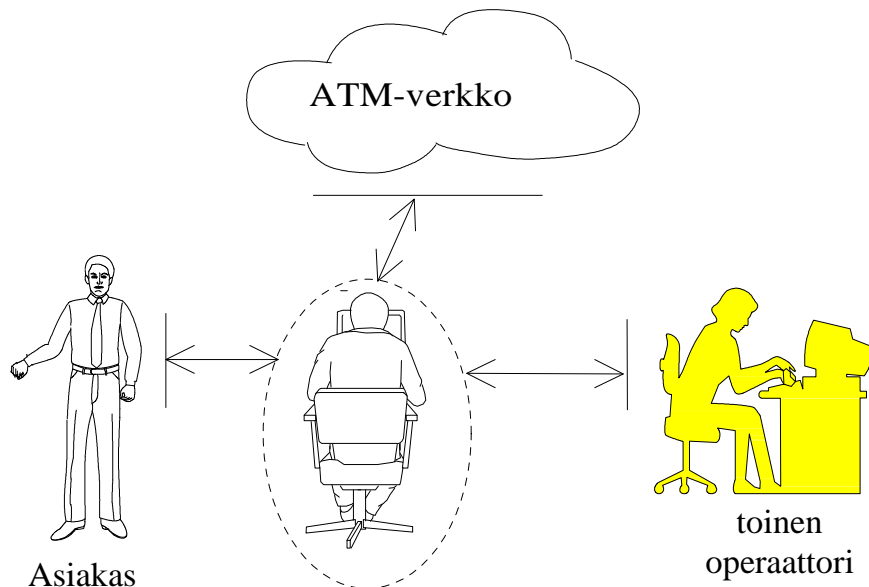
Tarkastelu lähtee liikkeelle verkko-operaattorin näkökulmasta, vieläpä niin, että lähtökohtana ovat operointia hoitavat henkilöt, "operaattorit". Operaattorien tehtävänä on hoitaa verkko siten, että asiakkaille voidaan tarjota hyvälaatuisia atm-verkon palveluita ja verkon kapasiteettia käytetään mahdollisimman tehokkaasti.

Mutta mikä on atm-verkko? Alkuvaiheessa riittänee määritelmä atm-verkolle asetettavista vaatimuksista (puuttumatta tarkemmin siihen miten tavoitteet saavutetaan):

- ⇒ Atm-tekniikan avulla on tarkoitus luoda tietoliikenneverkko, joka pystyy tarjoamaan kaikentyyppisille palveluille niiden tarvitsemat tietoliikenneyhteydet. Kyseistä tietoliikenneverkkoa kutsutaan tässä yhteydessä atm-verkoksi.

Näin määritellyn atm-verkon ominaisuudet muuttuvat ajan ja tekniikan kehittymisen myötä. Tällä hetkellä ollaan vaiheessa, jossa atm-verkko ei vielä pysty täyttämään sille asetettuja tavoitteita kokonaisuudessaan. Tietyt perusominaisuudet voidaan kuitenkin jo nyt tarjota asiakkaalle, kuten suuri siirtonopeus, joka riittää jopa huippulaatuisen televisiokuvan siirtämiseen, ja käytettävän siirtonopeuden vapaa määrittely jokaisen yhteyden tapauksessa erikseen. Tietyt, suurten verkkojen kannalta välttämättömät ominaisuudet, kuten valintaiset yhteydet, ovat vielä hyvin puutteellisia. Tämä esitys pyrkiikin tarkastelemaan tilannetta siinä vaiheessa kun vaatimukset ovat paremmin toteutettavissa, ehkä parin kolmen vuoden kuluttua.

Operaattorilla on kolmen tyyppisiä rajapintoja: asiakkaisiin, omaan verkkoon ja muiden verkkojen operaattoreihin (kuva 4.1). Kuvan tarkoituksena on siis lähinnä määritellä tämän selostuksen näkökulma (monien mahdollisten joukosta).



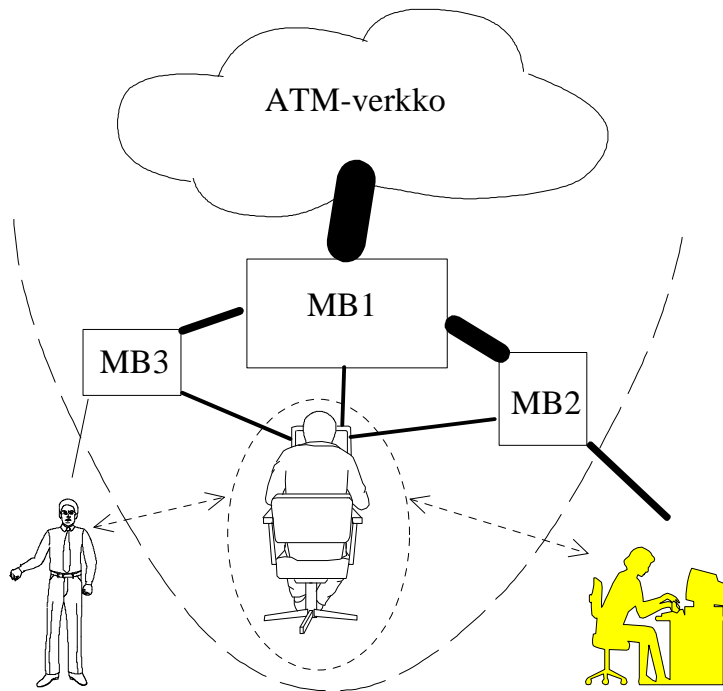
Kuva 4.1. Operaattorin yhteydet (1)

Tällä tavoin ajateltuna atm-verkon liikenteen hallinta on operaattorin käyttämä apuväline, joka mahdollistaa annetun tehtävän hoitamisen. Käytetään siis seuraavaa määritelmää:

- ⇒ Liikenteen hallinnan tavoite on taata verkon sujuva toiminta, siten että asiakkaat ovat tyytyväisiä verkosta saamiinsa palveluihin.

Toisaalta oletuksena on, että on olemassa henkilöitä, jotka viime kädessä vastaavat verkon toiminnasta, tekevät tarvittaessa verkon toimintaan vaikuttavia päätöksiä erityisesti ongelmatapauksissa, ja ovat yhteydessä jollain tavalla sekä asiakkaisiin että toisten verkkojen operaattoreihin.

Seuraava askel on määrittellä kuvan 1 avulla ne liikenteen hallinnan perusosat, joihin operoivalla henkilöllä on suora yhteys. Nämä on esitetty kuvassa 4.2 (MB1, MB2 ja MB3). Tärkein näistä on MB1, jonka päätetehtävä on jalostaa atm-verkon sisältämä suunnaton informatiomäärä ymmärrettävään muotoon. Osa tästä informaatiosta voidaan myös välittää muille operaattoreille ja asiakkaille. Operaattorin tehtävä on seurata näitä informatiovirtoja ja tarvittaessa myös vaikuttaa niihin. Näiden, tietokoneisiin perustuvien yhteyksien lisäksi on myös olemassa suuremmat kontaktit, joiden hoitamisen kannalta on hyvin oleellista, että operoivalla henkilöllä on mahdollisimman selkeä ja pätevä kuva verkon liikenteestä ja siihen vaikuttavista tekijöistä.

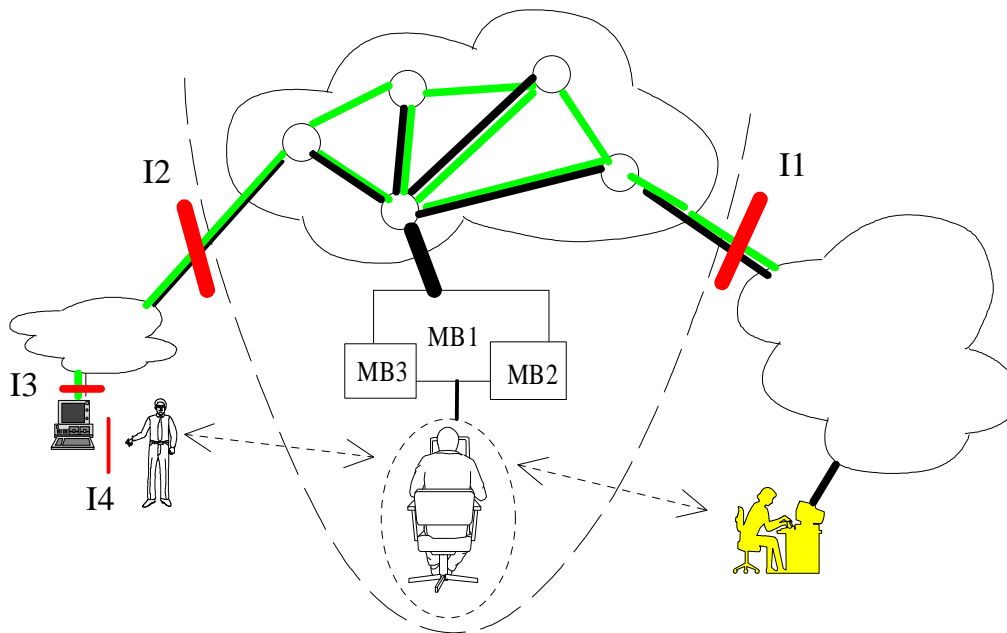


Kuva 4.2. Operaattorin yhteydet (2)

Tässä yhteydessä on ehkä hyödyllistä palauttaa mieleen se tosiasia, että investoinnin osuus jää tyypillisesti 30 prosenttiin hankinnan koko elinkaaren kustannuksista (kyseinen luku on esitetty hyvin samantyyppisessä asiayhteydessä, nimittäin vertailtaessa Ethernetin ja Token ringin kustannuksia¹). Loppuosaa kustannuksista muodostuu paljolti henkilökuluista, joista oleellinen osa liittyy verkon ja liikenteen hallintaan.

Johtopäätöksenä voidaan sanoa, että kaikki liikenteen hallintaan liittyvät toimenpiteet tulisi toteuttaa menetelmillä, jotka normaalitilanteessa eivät vaadi minkäänlaista operoijien puuttumista asiaan. Menetelmien toiminta (ja niihin liittyvät parametrien merkitys) tulisi kuitenkin olla ymmärrettäviä, jotta verkon valvonta ja ongelmatilanteiden ratkaiseminen olisivat käytännössä mahdollisia. Jos jokin hallinnan osa on erityisen vaikea ymmärtää, sen toiminnan tulee olla äärimmäisen luotettavaa myös poikkeuksellisissa olosuhteissa kuten ylikuormitustilanteissa. Lisäksi on huomattava, että asiakassuhteiden tehokas hoitaminen edellyttää, että asioita on voitava selittää myös vähemmän asiaa tunteville henkilöille, mieluummin ymmärrettävästi. Eli ainakin asiasta kiinnostuneilla henkilöille on oltava tarjolla verkon toimintaa kohtuullisen hyvin kuvaava malli.

¹K. Tolly: Token Ring vs. Ethernet: The Real Cost Story



Kuva 4.3. Operaattorin yhteydet (3).

Jatketaan kuvan täydentämistä. Käytännössä hyvin monimutkaista televerkkojen muodostamaa kokonaisuutta jäsennetään rajapintojen avulla. Operaattorin kannalta tärkeimmät ovat rajapinnat asiakkaan ja toisten operaattoreiden verkkoihin (kuvassa 3 rajapinnat I1 ja I2, huom. nämä eivät ole standardin mukaisia merkintöjä). Toisaalta suuntaus on menossa siihen, että julkisen verkon operaattori voi vastata myös yksityisen verkon operoinnista eli rajapintojen I2, I3 ja I4 välisestä alueesta. Julkisen verkon operaattori voi pitää huolta koko verkosta ja mahdollisesti päätelaitteissa pyörivistä sovelluksista ja pisimmilleen vietyä jopa käyttäjien koulutuksesta.

Teknisestä näkökulmasta katsottuna kuvan 3 rajapinnat riittänevät varsin pitkälle. Asia, eli palveluiden tarjonta asiakkaille, ei kuitenkaan ole puhtaasti tekninen. Kuvaan 3 tulee vielä lisätä yksi rajapinta, joka kulkee, niin sanoakseni, käyttäjän päässä. Tämä, hyvin kuvitteellinen rajapinta (I5) kuvaa sitä miten käyttäjä lopulta hyödyntää verkon avulla toteutettua palvelua. Tästä ilmiöstä otettakoon esimerkiksi videoneuvottelu: palvelusta on olennaista hyötyä vasta sitten, kun se on kulkenut tämän viimeisenkin rajapinnan läpi ja sitä kautta vaikuttanut johonkin ihmisessä olevaan prosessiin. Palvelun laatua tässä rajapinnassa voidaan kuvata Mean Opinion Score (MOS) -tyyppisillä käsitteillä. Vaikka MOS voikin antaa jossain määrin epämääräisiltä ja satunnaisilta vaikuttavia tuloksia, voidaan sen antamia tuloksia pitää perustavampaa laatua olevina kuin täsmällisempiä suureita, kuten bittivirhesuhde tai viive (rajapinnassa I3) tai videokuvan signaalikohinasuhde (rajapinnassa I4).

4.3 Rajapinnat ja parametrit

Edetään ensin kokonaiskuvan kehittämisen suuntaan ja tarkastellaan vasta tämän jälkeen yksittäisten osien toimintaa ja ominaisuuksia.

Oikeastaan kaikki lähtee liikkeelle rajapinnasta I5, eli asiakas haluaa käyttää jotain palvelua jonain hetkenä, ja tämän palvelun toteuttaminen vaatii tietyn tyyppistä tiedonsiirtoyhteyttä. Tiedonsiirtotarpeen perusteella generoituu sitten yhteyspyyntö, joka operaattorin kannalta merkitsee sen hallinnassa olevan verkon kahden rajapinnan välistä yhteyttä (esimerkiksi I2 ja I1 -tyyppisten rajapintojen välillä, ajatusta voi tietenkin laajentaa monimutkaisempaan yhteyksien kokonaisuuteen).

Tähän tapahtumakulkuun liittyy eräs ilmeinen hankaluus: asiakas haluaa palvelua, mutta operaattori on kiinnostunut siirtoyhteydestä. Tämä väli on täytettävä jollain tavalla.

Atm-verkon perusominaisuushan on siirtoyhteyksien ominaisuuksien varsin vapaa määrittely. Toisaalta tällä, sinänsä hyvällä ominaisuudella, on varjopuolensa, eli siirtoyhteyden ominaisuudet on *pakko* määritellä, sillä verkon järkevä toiminnan kannalta on välttämätöntä, että sekä käynnissä olevien että tarjolla olevan uuden yhteyden ominaisuudet tunnetaan. Tästä seuraa, että asiakkaan on yhteyttä muodostettaessa kerrottava yhteyden asettamat vaatimukset edes jollain tarkkuudella. Tietysti, mikäli mahdollista, tämän tulee tapahtua mahdollisimman pitkälle automaattisesti, jotta pitkää listaa liikenteen parametrejä ei joka kerta erikseen tarvitse pohtia.

Siirtoyhteydelle asetettavat vaatimukset sisältävät olennaisesta kolmen tyyppisiä tietoja: osa koskee suoranaisesti liikenteen ominaisuuksia, kuten tarvittava siirtonopeus, ja osa laatuparametrejä kuten virheellisten bittien osuus, ja lisäksi asiakas voi ilmaista käsityksensä sopivasta palvelun hinnasta. Merkitään näitä suureilla $P2^+$, $Q2^+$ ja $T2^+$, jossa yläindeksi (+) viittaa tässä suuntaan asiakkaasta verkkoon päin ja 2 viittaa rajapintaa I2.

Toisaalta asiakkaan kannalta on tarpeen, että verkko pystyy kertomaan minkälaista palvelua se pystyy yhteydelle tarjoamaan ($P2^-$, $Q2^-$ ja $T2^-$). Eli yhteyttä muodostettaessa tapahtuu jonkinlainen neuvottelu, jonka tuloksena on sopimus, jossa verkko lupaa tarjota tietynlaista palvelua ja asiakas lupaa käyttäytyä tietyllä tavalla ja maksaa ennalta sovittu korvaus yhteydestä.

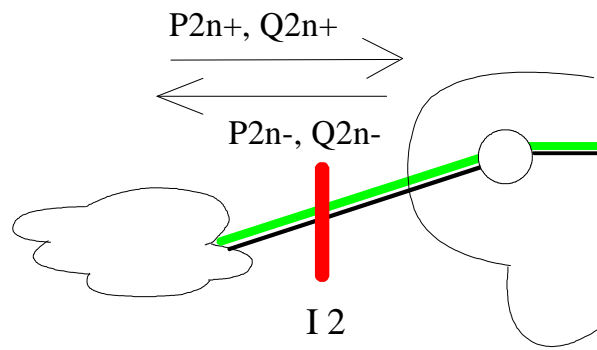
Tähän prosessiin liittyy useita vaiheita. Merkitään eri vaiheisiin liittyviä parametrejä merkinnöillä: n = neuvottelun aikana esitetyt tarjoukset, d = neuvottelun tuloksena sovitut arvot ja r = toteutuneet arvot.

Näin ollen päästään seuraavan tyyppisiin merkintöihin:

- $P2n^+$ = käyttäjän haluamat liikenteen parametrit rajapinnassa 2
- $T2d =$ sovitut hintaparametrit rajapinnassa 2, jolloin tulee päteä $T2d^+ = T2d^-$, jotta kyseessä on sopimus
- $Q2r^- =$ verkon tarjoaman palvelun toteutuneet laatuparametrit rajapinnassa 2

Vaikka nämä merkinnät eivät varmaankaan kata kaikkia mahdollisia tapauksia, ne riittävät useimpiin käytännön tarpeisiin, sillä asian monimutkaistamisellakin on omat haittapuolensa.

Verkon ja liikenteen hallinnan kannalta asian tekee monimutkaiseksi se, että parametrien määrittelyt ja arvot vaihtelevat sen mukaan mistä kohtaa yhteyttä, eli mistä rajapinnassa, informaatiovirtaa tarkastellaan. Rajapinnat I1, I2 ja I3 voivat olla, mutta eivät välttämättä ole, hyvin samankaltaisia. Näissä rajapinnoissa parametrit kuvaavat lähinnä verkon suorituskykyä, kun taas rajapinnassa I4 kyseessä on vastaavasti palvelun mitattavissa oleva laatu ja rajapinnassa I5 käyttäjän kokema palvelun laatu. Eri rajapintojen parametrien välillä on tiettyjä selkeitä riippuvuuksia: esimerkiksi jossain verkon osassa syntyvä suuri viive vaikuttaa myös käyttäjän kokemaan palvelun laatuun. Usein riippuvuussuhteet ovat valitettavasti varsin vaikeasti määritettävissä.



Kuva 4.4. Liitäntä ja sen kautta välitettävä informaatio yhteyttä muodostettaessa.

4.4 Palveluiden ominaisuudet eri rajapinnoissa

Käydään seuraavaksi läpi erilaisten palveluiden ominaisuuksia eri rajapinnoissa.

Rajapinta I5

Tässä on kyse siitä, mitä käyttäjät todella haluavat. Esimerkkejä kysymyksistä, joihin tulee etsiä vastauksia:

- Minkä tasoista puhetta ihmiset yleensä haluavat; riittääkö gsm-taso; kuinka paljon paremmasta laadusta ollaan valmiita maksamaan?
- Minkä tyyppinen ja -tasoinen palvelu riittää useimmille videoneuvotteluun osallistuville, jos toteutus perustuu erillisiin studioihin tai jokaisen omaan mikroon?
- Onko reaaliaikainen videokuva oleellinen Internetissä eli ollaanko siitä valmiita maksamaan?

Lukuarvojen käyttö ei ole tässä vaiheessa ole useinkaan hyödyllistä (?). Lähinnä voidaan käyttää MOS-arvoja, ja pyrkiä selvittämään niiden suhdetta siihen miten paljon käyttäjät ovat valmiita maksamaan palvelusta. Nämäkin riippuvuudet ovat hyvin vaikeasti selvitettävissä ja muuttuvat jatkuvasti. Yleensä käyttäjät haluavat mahdollisimman halpoja palveluita, joskin joissakin tapauksissa kalleus voi olla myös etu ("Rolls Royce" -ilmiö).

Rajapinta I4

I5:n ja I4:n välillä on selvästikin tiettyjä riippuvuuksia: kyse on siitä voidaanko I4:ssä havaittavista asioista tehdä malli, jonka avulla voidaan ennustaa hyvin I5-rajapinnassa havaittavan "todellisen" palvelun laatu. Tähän liittyviä kysymyksiä ovat esimerkiksi:

- Miten määritellään puheen laaduun kannalta oleelliset parametrit "ilmarajapinnassa", ainakin kaistanleveys on oleellinen, entä signaalikohinasuhde (miten määriteltynä ja minkälaiselle signaalille?)
- Mitkä ovat oleelliset kuvan laadun parametrit: signaalikohinasuhde(?), entä voidaanko kuvan nykivyyttä mitata jollain yksinkertaisella tavalla.

- Mitkä ovat Internet-palveluille ominaisia suureita, jotka eivät ole vain eri osien (teksti, kuva, ääni, video) ominaisuuksien summa?

Puheen siirto asettaa varsin tiukat vaatimukset viiveelle, nimittäin, jos yhteydellä ei käytetä kaikusalpoja tai vastaavia laitteita, sallittu viive on 100 ms luokkaa. Toisaalta kaikusalpoja käytetään tietyissä verkkoratkaisuissa joka tapauksessa, jolloin viive voi olla suurempikin ehkä 0.5 s luokkaa.

Puheen koodauksessa päästään varsin alhaisiin bittinopeuksiin ilman että puheen ymmärrettävyys kärsisi. Sen sijaan puheen sävyjen erottelukykyä on vaikeampi arvioida. Huomattakoon, että valtaosa normaalin keskustelun sisältämistä viesteistä (rajapinnassa I5) välittyy erilaisten sävyjen avulla, ei siis pelkkien sanojen muodostamana informaatiiovirtana. Jos siirtoyhteys kuljettaa pelkkiä sanoja mutta ei sävyjä, niin palvelun arvo voi kärsiä oleellisesti. Sävyjen parempi erottelukyky puoltaa myös laajemman taajuusalueen siirtämistä verkossa (esim. 0.1-7 kHz).

Mitkä ovat sitten ne tekijät, jotka määrittelevät videokuvan laadun erilaisissa käyttötarkoituksissa. Kuvapuhelin on oleellisesti eri asia kuin elokuvan katselu, mutta löytyykö joitain parametrejä, jotka soveltuvat laadun arviointiin yleisesti: kuvan koko, terävyys, liikkeen toisto (miten esimerkiksi kuvataan lukuarvolla sitä, että ihmisen kävely näyttää hieman nykivältä?). Tehtävä on ilmeisen vaikea, sillä jopa suhteellisen yksinkertaisen harmaasävykuvan tapauksessa asiantuntijan arvio on, että vaikka lukuisia numeerisia mittareita on kehitetty, niin yhdellä skalaarisuureella ei voida kuvata erilaisia kuvan heikkouksia².

Entä sitten data-siirto. Mitä ensiksikin datalla tarkoitetaan tässä yhteydessä? Määritelmäehdotus:

- ⇒ Datasiirtoa on kaikki se tietoliikenne, jossa informaatio ei kulje suoraan rajapintaan I5 saakka. Mutta miten tulkitaan sana "suoraan"; kyse on paljolti sallitusta viiveestä, rajan ollessa ehkä 1-2 sekunnin luokkaa.

Näin määriteltynä esimerkiksi videoservertä omalle levyille siirrettävä videopätkä on datasiirtoa. Sen katselua koskevat pääosin samat vaatimukset kuin "suorana lähetyksenä" katsottua videota rajapinnassa I4; sensijaan vaatimukset rajapinnoissa I1-I3 ovat oleellisesti erilaisia.

Lisäksi erilaiset sovellusten yhdistelmät, "multimedia-palvelut", voivat aiheuttaa ylimääräisiä vaatimuksia, jotka eivät suoraan seuraa erillisten sovellusten asettamista vaatimuksista. Tällainen on esimerkiksi puheen ja kuvan synkronoinnille asetettava varsin tiukka vaatimus.

Rajapinta I3

Tämä rajapinta voi olla (tai voi olla olematta) atm-pohjainen. Tämän tarkastelun kannalta on hyödyllisempää olettaa se muuksi kuin atm:ksi, koska tällaiseen verkkorakenteeseen liittyy joitakin erityisongelmia. Palvelua kuvaavia suureita I3:ssa ovat tyypillisesti:

P3: bittinopeus ja sen vaihtelu; mahdollinen pakettien koko ja koon jakauma,

Q3: virhesuhteet (bitti, kehys, sekunti jne.), viiveet ja viiveen vaihtelu.

Nämä parametrit vaikuttavat rajapinnan I4 palvelun ominaisuuksiin; joskaan parametrien suhde ei ole aina aivan helposti määriteltävissä.

²A. M. Eskicioglu, P. Fisher: Image Quality Measures and Their Performance, IEEE Trans. on Comm., Dec. 1995.

Voimakkaasti kompressoitu videokuva vaatii hyvin pienen todennäköisyyden sille, että kokonainen kehys menetetään. Lisäksi vaihtelevanopeuksinen videokuva on hankala toteuttaa, jos viiveen vaihtelu on suurta.

Datsiirto pystyy hyödyntämään usein hetkellisesti hyvin suuria siirtonopeuksia, vaikka pitkän ajan keskinopeus voi olla melko alhainen. Datsiirtoyhteyden asettama *vaatimusta* siirtoyhteyden nopeudelle on yleensä vaikea esittää yksiselitteisesti, joskin voitaneen määrittellä pienin käyttökelpoinen nopeus sekä suuremmista nopeuksista saatava hyöty jollain tavoin.

Liityntäverkko (rajapintojen I2 ja I3 välinen osuus) voi olennaisesti vaikuttaa sovelluksen liikenteellisiin ominaisuuksiin julkisen verkon rajapinnassa. Esimerkiksi ethernetin tai gsm-verkon kautta tulevien puheluiden liikenteelliset ominaisuudet voivat poiketa toisistaan olennaisesti, joten liikenteen hallinnalla on oltava jonkinlainen käsitys liityntäverkosta, jotta liikenteen hallinta, ja sitä kautta myös verkon resurssien käyttö, olisi tehokasta. Tämä taas on välttämätöntä, koska käyttäjät vaativat edullisia hintoja.

Rajapinta I2

Tämä rajapinta voidaan olettaa atm-pohjaiseksi. Olennaisimpia tietoja liikenteen hallinnan kannalta ovat:

- P2: tarvittava solunopeus ja sen tilastolliset ominaisuudet sekä mahdollisuus sovittautua kulloisenkiin käytettävissä olevaan nopeuteen,
- Q2: sallittu solujen menetystodennäköisyys, sallittu viive ja sallittu viiveen vaihtelu.

Nämä ovat selvästikin sellaisia suureita, joiden esitystapa täytyy määrittellä jossain standardissa yksityiskohtaisesti, jotta eri valmistajien laitteita voitaisiin käyttää eri osissa verkkoa. Avoimia kysymyksiä on lukuisia; niitä tarkastellaan lähemmin seuraavassa luvussa.

Rajapinta I1

Tämä rajapinta voidaan olettaa atm-pohjaiseksi. Liikenteen parametrit ovat hyvin pitkälle samat kuin rajapinnassa I2. Tämän rajapinnan erityisongelmat liittyvät lähinnä siihen, että rajapinnan vastakkaisilla puolilla olevat operaattorit ovat useinkin jollain tasolla toistensa kilpailijoita, joten välitettävä informaatio voi olla puutteellisempaa kuin se mitä on käytettävissä rajapinnassa I2.

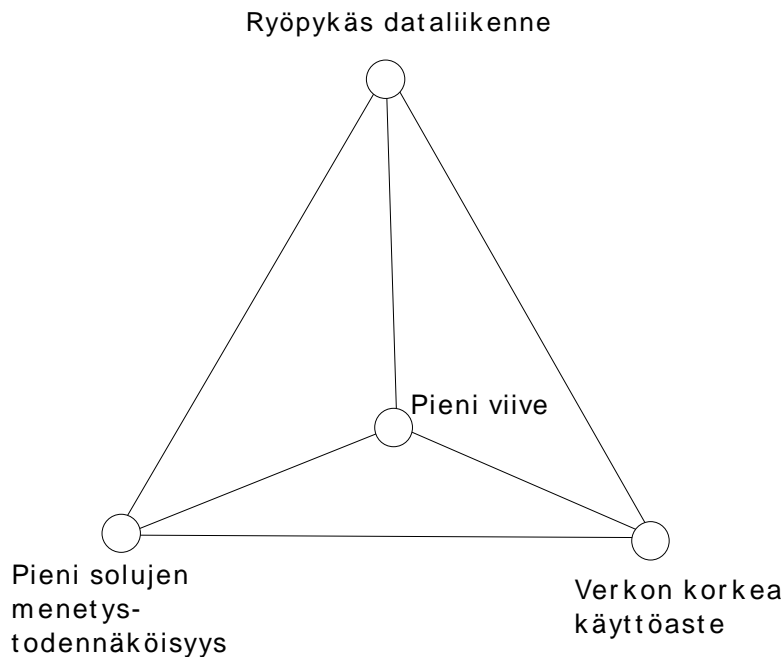
4.5 ATM-verkon operoinnin perusongelmia

ATM-tekniikkaan perustuvien tietoliikenneverkkojen kokonaisuuteen liittyy lukemattomia, osin vielä lopullista vastausta vailla olevia kysymyksiä ja ongelmia. Seuraavassa käydään läpi joitakin tärkeimmiksi katsottavia ongelmia, ja esitetään joukko erilaisia ratkaisumalleja.

Dataliikenne ja palvelun laadun takaaminen

Erilaiset atm-verkon liikenteen hallinnalle asetettavat vaatimukset ovat valitettavasti keskenään ristiriitaisia. Hyvin lyhyesti sanottuna: pientä viivettä, pientä solujen menetystodennäköisyyttä ja korkeaa verkon käyttöastetta ei voida saavuttaa yhtä aikaa, jos merkittävä osa liikenteestä on

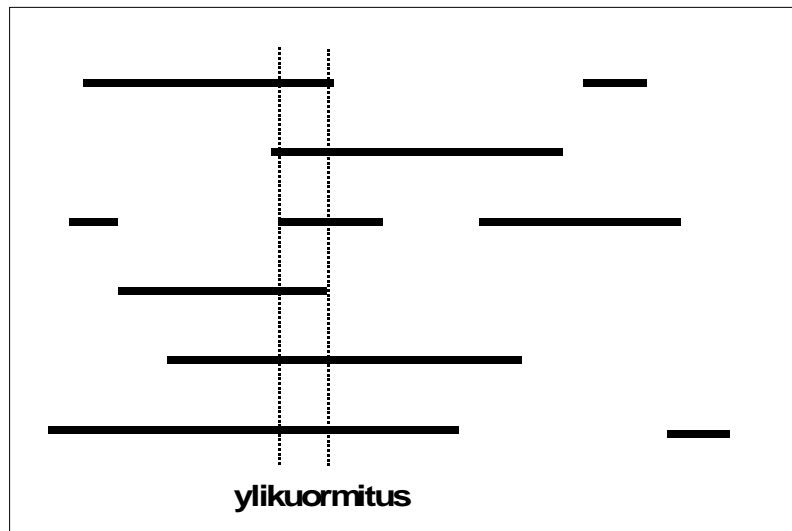
voimakkaasti vaihtelevaa dataliikennettä. Tätä ongelmaa havainnollistavat kuvat 4.5 ja 4.6. Alunperin kuviteltiin (ainakin melko yleisesti), että atm mahdollistaa samanaikaisesti sekä pienen viiveen, pienen solujen menetystodennäköisyyden että korkean verkon käyttöasteen. Tämä pitääkin paikkansa, jos liikenteen ominaisuudet täyttävät tietyt vaatimukset. Jos yhteydet ovat tyypiltään vakionopeuksisia (CBR, constant bit rate) niin 100 solun kokoisilla puskureilla päästään noin 90 % kuormitukseen pahimmassakin tapauksessa (eli kun kaikkien yhteyksien nopeus on hyvin pieni) ja samalla voidaan taata, että solujen menetystodennäköisyys on alle 10^{-9} . On huomattava että puskurin koko määrää suoraan viiveen vaihtelun maksimi-arvon (jos kyseessä on lähtöpuskurointi), joka tässä tapauksessa on noin 0.27 ms 155 Mbit/s väylällä.



Kuva 4.5. Vaatimusten ristiriitaisuus atm-verkossa³

Mutta entä jos liikenne ei ole vakionopeuksista vaan tyypiltään vaikkapa on/off. Tällöin on erittäin oleellista se, kuinka suuren osan jaettavissa olevasta siirtokapasiteetista yksi yhteys varaa. Jos tämä osuus on suuri, niin tilastollinen kanavointi ei ole (sellaisenaan) mahdollista, jos pienestä viiveestä tai pienestä solujen menetystodennäköisyydestä ei luovuta. Tätä ilmiötä on pyritty havainnollistamaan kuvalla 4.6.

³Kuvan on esittänyt prof. J. Virtamo virkaanastujaisesitelmässään 18.9.1995



Kuva 4.6. Liikennetarve ajan funktiona; jokainen yhteys varaa 1/5 väylän kapasiteetista

Oletetaan, että väylällä on 6 on/off yhteyttä, joista kukin varaa 1/5 käytettävissä olevasta kapasiteetista. Tällöin sattuu kohtuullisen suurella todennäköisyydellä tilanteita, joissa jokainen lähde haluaa lähettää yhtä aikaa. Ongelma on pakko ratkaista tavalla tai toisella. Erilaisia mahdollisuuksia on useampiakin (oikeastaan tässä on hahmoteltu erilaisia atm-verkon palveluita hyvin vaihtelevanopeuksiselle dataliikenteelle):

1. Tasoitetaan liikenteen vaihtelut jo ennen varsinaista atm-verkkoa. Tällainen ratkaisu voi vaatia tilaajaliitännässä hyvin suuria puskureita, koska on- ja off-tilojen kestot ovat pitkiä, ja samalla voidaan aiheuttaa kohtuuttoman suuria viiveitä. Varsinaisen atm-verkon kannalta tilanne saadaan näyttämään hyvältä, koska yhteydet ovat (tai niiden voidaan olettaa olevan) vakionopeuksisia, jolloin verkon (näennäinen) käyttöaste saadaan korkeaksi, mutta käyttäjän kannalta tilanne saattaa olla sangen epäedullinen.
2. Pyritään välttämään ylikuormitustilanteiden syntyminen, eli hylätään yhteys jo ennalta, jos takeita kapasiteetin riittävydestä ei voida antaa. Käytännössä tämä johtaa huippunopeuden mukaiseen allokointiin, jos yhteyksien huippunopeudet ovat suuria. Samalla tietysti väylän käyttöaste saattaa jäädä hyvin alhaiseksi.
3. Tehdään kapasiteetin riittävyden tarkistus (Connection Admission Control, CAC) erikseen jokaiselle ryöpylle (Fast Reservation Protocol/Delayed Transfer). Tämä aiheuttaa tiedon siirtoon verkon koosta riippuvan lisäviiveen, koska kapasiteetin riittävyden tarkistamiseen kuluu aina jokin aika. Huomattakoon lisäksi, että osa kapasiteetista menee hukkaan, koska sen varaus on tehtävä myös siksi aikaa, kun tieto vapaasta kapasiteetista saadaan käyttäjälle saakka.
4. Edellisen kohdan kapasiteetin tarkistus voidaan tehdä myös siten, että ryöpyt lähetetään matkaan välittömästi (Fast Reservation Protocol/Immediate Transfer -menetelmä), jolloin ei voida taata solujen läpimenoa. Lopputulos on käytännössä se, että joko kaikki solut tai ei yhtään solua menee onnistuneesti läpi. Eli ryöppy voidaan joutua lähettämään uudelleen, jos jossakin kohtaa verkkoa ei ole riittävästi tilaa. Sekä 3. että 4. ratkaisu edellyttävät käyttäjän päässä riittävän puskuritilan, jotta tiedonsiirto voidaan siirtää sopivaan ajankohtaan.
5. Ylikuormitustilanteen aikana jokainen yhteys menettää solujaan (sen jälkeen kun puskuri on täyttynyt). Viiveet voidaan pitää pienenä käyttämällä pieniä puskureita.

Tämä ratkaisu voi johtaa hyvin korkeaan solujen menetystodennäköisyyteen ja käyttäjien saama hyöty ylikuormitustilanteessa voi olla hyvin vähäinen. Tällöin, sovelluksesta riippuen, tilanne saattaa olla sama kuin, jos yhteys olisi osan aikaa poikki.

6. Voidaan kehittää älykkäämpiä sääntöjä sille, mitkä solut hylätään, jos tilanne sitä vaatii. Perinteinen tapa on käyttää atm-solun CLP-bittiiä, jonka avulla erotellaan tärkeät ja vähemmän tärkeät solut. Valitettavasti pelkän datasiirron tapauksessa tämä ratkaisu ei liene mielekäs, koska kaikki informaatio on yleensä tärkeää.
7. Edellisissä ratkaisumalleissa (1-6) on voitu olettaa, että verkon puskurit ovat pieniä. Jokaiseen atm-verkon solmuun voidaan sijoittaa myös hyvin suuret puskurit, joita sitten täytellään ylikuormitustilanteissa. Valitettavasti atm-verkossa käytettävistä hyvin suurista nopeuksista johtuen puskurin täytyy olla suunnaton, jos luotetaan pelkästään puskurointiin ja verkon käyttöaste pyritään pitämään korkealla. Samalla tietysti viive ja sen vaihtelu kasvavat voimakkaasti. Datasiirron tapauksessa on yleensä parempi ratkaisu se, että vain yksi (tai tarvittaessa useampi) yhteys menettää solujaan, kun taas muut yhteydet saavat kaikki solunsa läpi. Ongelmana on valita oikeudenmukaisesti ne yhteydet, jotka menettävät soluja. Yleensä on edullisinta toteuttaa protokolla siten, että vältetään yksittäisten solujen menetykset, koska yhdenkin solun menettäminen voi aiheuttaa suuren paketin uudelleen lähettämisen.

Eli menettäjä on voi olla yhteys,

- A. joka jo menettänyt yhden solun kyseisestä kehuksesta, tai
- B. lisäksi tietyn puskurin täyttöasteen jälkeen kaikki uusien kehysten solut hylätään tai
- C. voidaan ottaa huomioon myös se, kuinka paljon kukin yhteys on käyttänyt verkon resursseja.

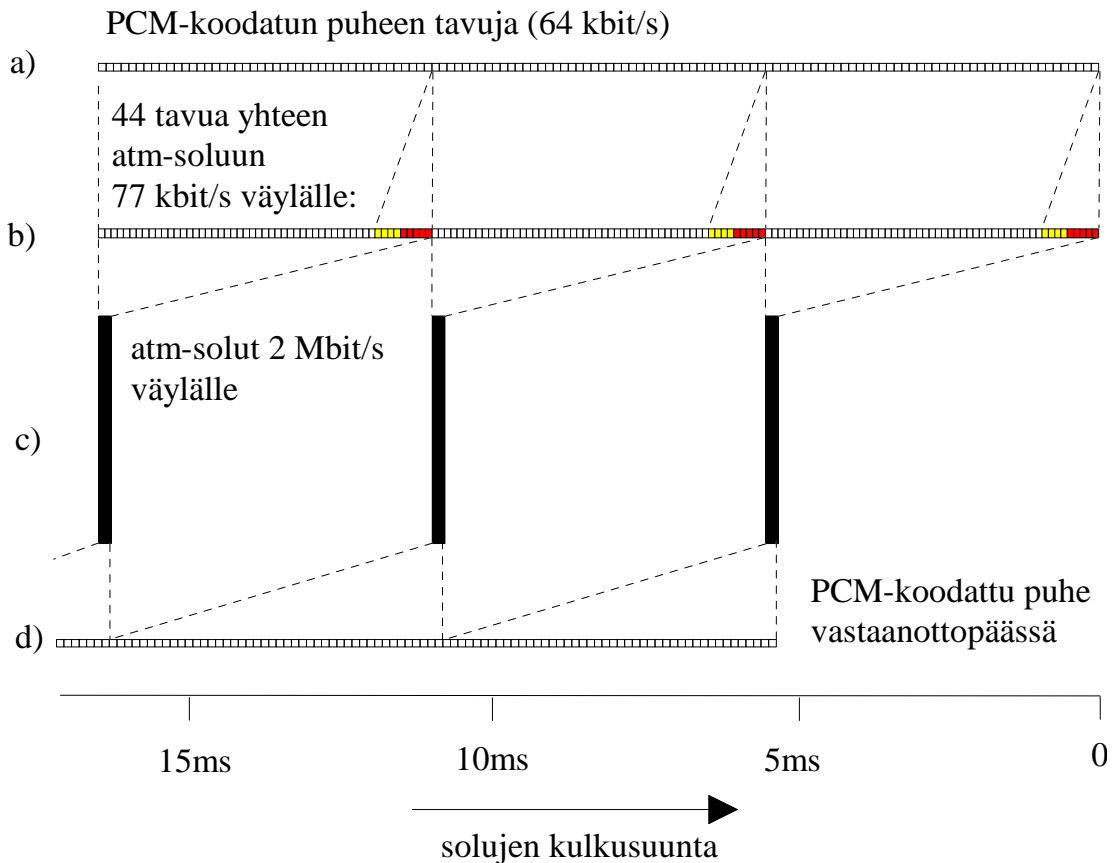
Nämä ovat Unspecified Bit Rate (UBR) -palvelun mahdollisia toteutustapoja. Oleellista tällä palvelutyypillä on se, että atm-verkko ei pyrikään takaamaan solujen varmaa läpimenoa, vaan verkko olettaa, että ylemmät protokollakerrokset havaitsevat epäonnistumiset ja toimivat sen jälkeen asianmukaisella tavalla.

8. Lähetetään tieto jokaiselle käyttäjälle siinä vaiheessa, kun ylikuormitustilanne on alkanut. Ongelma on jälleen se, että soluja ehtii kertyä ylikuormittuneeseen solmuun suuri määrä ennenkuin tieto on kulkeutunut kaikille asiakkaille, ja sitä kautta kuormitusta on saatu laskettua. Puskureiden on tässäkin tapauksessa oltava suuria, joten tällainen palvelu ei sovi viiveherkille sovelluksille. Solujen menetyksiltä saatetaan kuitenkin välttyä. Kyseisestä ratkaisusta käytetään nimitystä ABR (Available Bit Rate).

Eli yleisenä johtopäätöksenä voidaan sanoa, että aina joudutaan luopumaan jostain: joko pienestä viiveestä, pienestä solujen menetystodennäköisyydestä, verkon korkeasta käyttöasteesta tai siitä että verkkoon sallitaan kaiken tyyppisiä yhteyksiä. Kuvaan 5 sijoitettuna tämä merkitsee sitä, että käytännössä toteutettavat ratkaisut sijoittuvat aina tasolle, joka kulkee kolmen kuvassa esitetyn pisteen kautta.

Puheen välittämisen perusongelmat atm-verkossa

Edellä kuvattu koski siis lähinnä dataliikenteen hallintaan atm-verkossa. Puheen välityksessä atm-verkon yli on eräs perusongelma, joka muiden sovellusten yhteydessä ei yleensä nouse esille. Yleensä ajatellaan, että atm-verkko tarjoaa juuri sen määrän siirtokapasiteettia kuin sovellus tarvitsee. Tietystä aikaskaalassa tämä pitää kyllä paikkansa, mutta se ei ole koko totuus. Kuva 4.7 pyrkii selvittämään tätä väitettä.



Kuva 4.7. PCM-puheen välittäminen atm-soluilla: a) alkuperäiset tavut, b) 77 kbit/s väylä, 1 solu sisältää 44 tavua puhetta, c) 2 Mbit/s väylä, d) PCM-koodattu puhe vastaanottopäässä.

Ensimmäinen tavu oikeassa reunassa, korkeus kuvaa sitä miten nopeasti kukin tavu siirretään.

Kuvan 4.7 kohdassa *c* on esitetty "todellinen" yhteyden käytettävissä oleva kapasiteetti silloin kuin atm-yhteyden nopeus on noin 2 Mbit/s eli noin 30 kertaa yhden PCM-kanavan vaatima nopeus. Vastaavasti 155 Mbit/s väylällä piikki olisi vielä n. 75 kertaa korkeampi. Käytettävissä oleva nopeus vaihtelee siten itse asiassa äärimmäisen paljon vaikka sovellus vaatisi vakionopeuden.

Nyt voimme tietysti kysyä mikä on tästä aiheutuva ongelma, kun nopeusvaihtelu on varsin helposti tasoitettavissa vastaanottopäässä. Varsinainen ongelma ei olekaan nopeuden vaihtelu vaan se, että soluja ei ole tapana lähettää rinnakkain vaan peräkkäin silloin, kun väylän kapasiteetti riittää useamman samanaikaisen yhteyden välittämiseen. Eli jos 64 kbit/s puhe muutetaan atm-soluiksi ja solut lähetetään noin 77 kbit/s nopeuksiselle väylälle, ei välttämättä syntyisi juuri minkäänlaista viivettä. Tämä edellyttää, että atm-solua aletaan lähettää sitä mukaa kun informaatiota kertyy, eikä odoteta koko solun täyttymistä. Vastaavasti vastaanottopäässä informaatio lähetetään eteenpäin heti kun sitä on saatavilla. Ylimääräinen viive jää tällöin hyvin pieneksi (otsikkokenttien sijoittuminen informaation väliin aiheuttaa tietyn viiveen).

Entä sitten kun välillä on nopeampia väyliä, joilla useita, ellei lukemattomia yhteyksiä kanavoidaan samalla väylälle (muutenhan atm:n käytöstä ei ole mitään hyötyä). Solut on pakko siirtää tällöin

peräkkäin, jolloin solun aloitus pakosti siirtyy eteenpäin; käytännössä solun lähetys voidaan aloittaa vasta kun kaikki tarvittava informaatio on kasassa (vertaa kuvan tapauksia b ja c). Jokaisen solun ensimmäiset tavut viivästyvät siten vähintään sen ajan kun solun kokoaminen kestää. Jos soluun sijoitetaan esim. 44 tavua puhetta⁴, ja puheen koodausnopeutena käytetään 64 kbit/s, niin syntyvä viive on 5.5 ms. Koska puheen lähettämiseen on käytettävä täsmälleen samaa nopeutta kuin lähetettäessä, kyseinen viive koskee tietysti kaikkia tavuja.

Tämä viive, 5.5 ms, ei vielä ole kovin ongelmallinen, vaikeuksia alkaa syntyä vasta kun puheen koodausnopeus pienenee. Gsm-verkossa käytetään esimerkiksi nopeutta 13 kbit/s: vastaava viive on tällöin 27 ms. Mutta pienempiäkin nopeuksia on mahdollista käyttää ilman että puheen laadusta joudutaan juurikaan tinkimään. Tällöin viive alkaa jo olla ongelma, varsinkin kun verkon rakenne saattaa olla sellainen, että kyseinen viive kertautuu useamman kerran ja viivettä syntyy muutenkin. Ongelma on tiedossa ja siihen on olemassa joitakin ratkaisuja:

1. Käytetään kaiunpoistajia tai vastaavia laitteita siten, että käyttäjän kokema häiriö voidaan minimoida⁵. Esimerkiksi gsm-verkossa kaiunpoisto on vakiotoiminne (?). Erillislaitteiden sijoittaminen verkkoon voi tulla varsin kalliiksi, joten kaikkiin tilanteisiin tämä ratkaisu ei sovellu. Lisäksi on huomattava, että viiveen ei voida yleensä antaa kasvaa yli 400 ms (?), koska sen ylittävät arvot häiritsevät normaalia keskustelua, tosin Internetin yhteydessä 700 ms on pidetty hyväksyttävänä arvona⁶.
2. Ei käytetä "liian" pieniä nopeuksia. Tämä taas ei ole oikein järkevä ratkaisu silloin kun atm:ää käytetään nimenomaan sen takia, että se mahdollistaa pienten nopeuksien (ja puheeseen syntyvien taukojen) hyödyntämisen, silloin siirtokapasiteetti on kallis tai muuten rajallinen.
3. Ei täytetä soluja kokonaan. Tähän pätee paljolti edellisen kohdan kritiikki; lähinnä tämä tulee kyseeseen silloin kun tekniikka (kuten gsm) määrää koodausnopeuden ja jokin muu ehto määrää maksimiviiveen. Tällöin solujen vajaatäyttö saattaa olla käytännössä ainoa vaihtoehto.
4. Kanavoidaan useampia puhekanavia samaan yhteyteen. Tästä on kysymys silloin kun vaikkapa 30-kanavainen pcm-järjestelmä välitetään sellaisenaan atm-verkon yli. Atm-verkonhan ei tarvitse välttämättä tietää järjestelystä oikeastaan mitään. Tähän ratkaisuun liittyy ainakin kaksi oleellista ongelmaa: verkon sisällä ei voida tehdä mitään yksittäisiä puheluja koskevia kytkentöjä purkamatta koko yhteyttä, koska atm-verkko ei tunne solujen informaatiokentän jakamista. Toiseksi tilastollinen kanavointi on mahdollista vain kyseisen yhteyden sisällä, jos sielläkään. Järjestelmähän on tehokkaimmillaan silloin kun puhelujen määrä vastaa täsmälleen kanavan kapasiteettia, tai on sen monikerta. Samoin puheen taukojen hyödyntämistä ei ole kovin tehokasta jos jaettavana oleva kapasiteetti on pieni.
5. Jaetaan solut pienemmiksi osiksi. ATM-Forumin puitteissa on mm. ehdotettu atm-solun jakamista tarvittaessa kolmeksi minisoluksi, joilla jokaisella on pieni oma otsikkonsa. Viiveongelma ratkeaa tällä tavoin varsin kätevästi, mutta samalla

⁴44 tavua on käytetään esim. MFS Datanetin palvelussa, Data Comm., Oct. 1995

⁵ Samainen MFS Datanet, jossa kaiunpoisto tehdään Magellan-keskuksessa.

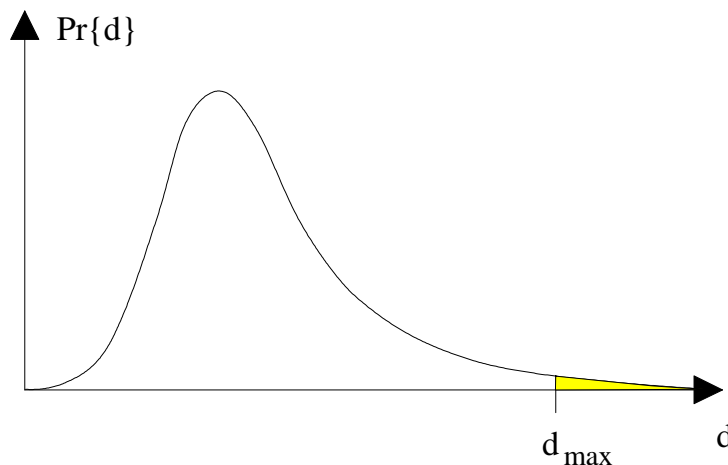
⁶Phoning across The Internet, Communications Int., Dec. 1995

syntyy muita ongelmia. Lähinnä kysymys on siitä, että ratkaisu vaatii atm-verkon tekniikalta ja liikenteen hallinnalta olennaisesti uusia ominaisuuksia, jos pyritään olennaisesti “parempaan” ratkaisuun kuin 4 kohdassa. Ratkaisusta saatava hyöty ei kuitenkaan ole N-kertainen verrattuna vajaiden solujen käyttöön (jossa N minisolujen määrä atm-solussa), sillä solujen jakaminen vaatii ylimääräisiä otsikkotietoja, eikä aina ole mahdollista täyttää jokaisen solun kaikkia minisolupaikkoja.

VBR-videon erityisongelmat atm-verkossa

Viiveen vaihtelu (Cell delay variation, CDV)

Videokuvan ja erityisesti vaihtelevanopeuksisen, siirto vaatii mahdollisimman pientä viiveen vaihtelua. Tämä johtuu toisaalta synkronoinnin vaikeudesta jos viiveen vaihtelu on suurta ja toisaalta siitä, että pyritään välttämään hyvin suuria puskureita vastaanottopäässä. ATM-verkko aiheuttaa kuitenkin luonnostaan viiveen vaihtelua, jonka suuruus riippuu hyvin monista tekijöistä. Käytännössä sovelluksen kannalta olennaista on se, miten suuri on viiveen vaihtelun maksimi-arvo. Koska absoluuttista maksimia on yleensä mahdotonta määrittellä, niin viiveen vaihtelun maksimi määritellään yleensä kuvan 4.8 mukaisesti arvona d_{\max} siten, että $\Pr\{d > d_{\max}\} < 1e^{-9}$. Tämän todennäköisyyden tulee olla korkeintaan samaa tasoa kuin solujen menetystodennäköisyydelle määritelty tavoitearvo, sillä jos solu viivästyy enemmän kuin d_{\max} , siitä ei ole yleensä enää mitään hyötyä, joten se voidaan katsoa menetetyksi. Näin määritelty d_{\max} pyrkii takaamaan sen, että vastaanottopäässä viiveen vaihtelut voidaan tasoittaa järkevällä tavalla.



Kuva 4.8. Viiveen vaihtelun jakauma

Miten sitten päästään hyvin pieneen viiveen vaihteluun atm-verkossa, jossa osa liikenteestä voi olla erittäin voimakkaasti vaihtelevaa ja osa verkon puskureista voi olla hyvin suuria; puskurin kokohan periaatteessa määrää hyvin pitkälle viiveen maksimi-arvon (joskin myös atm-solmujen kytkentäkentän rakenne ja jonotusjärjestelmä vaikuttavat tähän arvoon).

Ensiksikin on aivan välttämätöntä, että hyvin pientä viiveen vaihtelua vaativat yhteydet voidaan erottaa sellaisista yhteyksistä, jotka sietävät, ja usein myös aiheuttavat verkon sisällä, suurta viiveen vaihtelua. Ratkaisumallit voidaan jaotella seuraavasti:

1. Käytetään niin pieniä puskureita, että puskuroinnin aiheuttama viive ei voi nousta suureksi. Tällöin kuitenkin kyseisen puskurin kuormitus täytyy pitää hyvin alhaisena, jotta solujen menetystodennäköisyys ei nousisi liian korkeaksi. Jos tätä menetelmää aiotaan soveltaa kaikille yhteyksille riippumatta liikenteen

ominaisuuksista, liikenteen hyväksymisen on perustuttava huippunopeuteen, joten verkon kuormitus voi jäädä erittäin alhaiseksi.

2. Yleensä vbr-video tai muut vastaavat viiveen vaihtelun kannalta kriittiset palvelut eivät ole ainoita liikennelähteitä, eikä verkon kuormitusastetta voida jättää äärimmäisen alhaiseksi, joten yhteyksien erottelu on käytännössä välttämätöntä. Kyseessä on tällöin yhteyksien priorisointi eli jokaiselle palveluluokalle (tai jopa jokaiselle yhteydelle) on oma puskurinsa. Näiden puskureiden sisältöä lähetetään eteenpäin prioriteettien mukaisesti, jolloin osalle yhteyksiä voidaan taata parempi palvelun taso kuin muille; tässä tapauksessa pienempi viive ja viiveen vaihtelu.
3. Ääritapaus edellisestä kohdasta on sellainen, että jokaisella yhteydelle varataan kiinteä määrä puskuri- ja siirtokapasiteettia, ja jokainen verkon solmu pitää huolen, että jokainen yhteys saa varmuudella kyseiset kapasiteettiarvot. Käytännössä tarvitaan virtuaalikanavakohtaiset puskurit ja Round Robin -tyyppinen puskureiden tyhjennysjärjestelmä. Vaadittava puskurin kokonaiskapasiteetti saattaa kuitenkin nousta äärimmäisen suureksi.
4. Kuten edellä, mutta kunkin yhteyden tarvitsema kapasiteetti määritetään jatkuvasti. Tämä vaatii, puskurointijärjestelmän lisäksi, varsin monimutkaisen merkinannon, jotta jokainen solmu pysyy perillä jatkuvasti jokaisen yhteyden tarvitsemasta kapasiteetista.
5. Erillisten puskureiden käyttäminen kaikille eri palvelutyypeille ei ehkä ole aivan välttämätöntä, vaikka se onkin monessa suhteessa joustavin ratkaisumalli. Eräs mahdollisuus, jonka avulla voidaan yksinkertaisella tavalla yhdistää kaksi erilaista luokkaa on seuraava. Oletetaan, että reaaliaikaisuutta ja pientä solujen menetystodennäköisyyttä vaativille yhteyksille on oma puskurinsa, jonka koko on suhteellisen pieni, korkeintaan muutama sata solua. Kyseisen puskurin kautta kulkevat yhteydet jaetaan kahteen luokkaan: normaalit ja hyvin pientä viiveen vaihtelua vaativat yhteydet. Jos tuleva solu kuuluu normaaliin yhteyteen, solu sijoitetaan ensimmäiseen vapaaseen puskuripaikkaan. Sensijaan muut solut sijoitetaankin lähelle puskuri loppupäätä olevaan paikkaan tai ensimmäiseen kyseisen paikan jälkeiseen tyhjiin paikkaan. Jos hyvin pientä viiveen vaihtelua vaativien yhteyksien muodostama kuormitus on suhteellisen pieni, niille voidaan taata hyvin pieni viiveen vaihtelu. Keksintöilmoitus on jätetty, joskaan en osaa sanoa, miten käyttökelpoinen esitetty ratkaisu todellisuudessa on.
6. Käytetään erillisiä laitteita, joilla jokaisen virtuaalikanavan liikennettä tasoitetaan (aina tarvittaessa) verkon sisällä niin paljon kuin mahdollista. VBR-tyyppiselle yhteydelle tämä on kuitenkin hankalaa, koska "oikeaa" solujen välistä aikaa ei yleensä tunneta, silloin kun hetkellinen nopeus jää huippunopeuden alle. Käytännössä tasoitus voidaan siis tehdä erillisellä laitteella vain huippunopeuden aikana.

Liikenteen parametrien määrittäminen

Vaihtelevanopeuksiseen videokuvaan liittyy eräs toinen hyvin hankala ongelma, joka saattaa uhata koko palvelutyyppin käyttöönottoa. Nimittäin liikenteen parametrien määrittäminen ennen yhteyden muodostamista (tässä on siis kyseessä P2n ja P2r parametrien välinen ero). Kuten jo edellä todettiin, viiveen vaihtelu VBR-videon tapauksessa on syytä pitää hyvin pienenä. Tämä merkitsee sitä, että jos kapasiteettia ei jonakin hetkenä ole riittävästi, niin soluja ei voida kovinkaan paljon puskuroida.

Yhden videoyhteyden tarvitsema kapasiteetti on taas yleensä tilastollinen suure, eli siitä voidaan sanoa jotain todennäköisyysjakauman avulla. Tällaiseen tietoon perustuvia suureita on hyvin hankala valvoa ja niiden avulla on myös hyvin hankala antaa mitään varmoja lupauksia palvelun laadusta.

Ratkaisumalleja on taas useita:

1. Ei käytetä vaihtelevanopeuksista koodausta vaan vakionopeuksista eli CBR:ää. Tällöin huippunopeus on yleensä tunnettu eikä sen valvonta aiheuta ongelmia. Samalla viiveen vaihtelun aiheuttamat ongelmat ovat yleensä helpommin ratkaistavissa kuin VBR yhteyden tapauksessa.
2. Jos koodaus on jostain syystä vaihtelevanopeuksista, varataan yhteyden ajaksi kuitenkin vakionopeuksinen riittävän leveä kaista, mutta lähetetään soluja vain sen mukaan kuin lähde niitä tuottaa. Tämä voi olla kohtuullisen järkevää, jos yhteyden veloituksessa otetaan huomioon todellisuudessa käytetyn solut, eikä ainoastaan varattua nopeutta.
3. Kuten 2 kohdassa, mutta lähetetään verkkoon soluja huippunopeuden mukaan. Tämä saattaa olla käyttäjän kannalta edullista, jos viiveen vaihtelu on kriittinen tekijä (vertaa kohta 5 viiveen vaihtelun yhteydessä) ja yhteyden veloitus perustuu ainoastaan ennalta varattuun nopeuteen. Ylimääräisiä soluja voidaan tietysti yrittää hyödyntää jotenkin, esimerkiksi datan siirtoon.
4. Parametrien määrittäminen liikennemittausten perusteella. Tämä edellyttää yleensä sitä, että yhteydet voidaan ryhmitellä rajalliseen määrään luokkia. Kustakin luokasta hankitaan sitten tilastollista dataa mahdollisimman paljon ja tämä datan pohjalta määritellään verkon kannalta optimaaliset liikenteen parametrien arvot (mitattu liikenteen keskiarvo ei välttämättä ole optimaalinen parametrin arvo, jos yhteyksien ominaisuudet vaihtelevat suuresti). Perusongelma tässä on se, että luokittelu saattaa olla hyvinkin hankalaa: esimerkiksi www-yhteyden ominaisuudelle lienee lähes turha yrittää mitata keskimääräisiä suureita. Toisaalta jos erilaisia luokkia on hyvin suuri joukko, jää mittaustulosten tarkkuus heikoksi.
5. Myös reaaliaikaisia liikennemittauksia voidaan soveltaa. Tämän ratkaisumallin vaikeus piilee mittausjakson valinnassa: jos mittausjakso on lyhyt niin tuloksen (tilastollinen) tarkkuus on huono, jos taas mittausjakso on pitkä, niin liikenneprosessia ei voida enää olettaa stationääriseksi (eli ajasta riippumattomaksi). Hankalien pitkän aikavälin riippuvuuksia sisältävien liikenneprosessien tapauksessa, joihin myös vaihtelevanopeuksinen video kuuluu, mittaukseen jää pakosti melko suuri virhemarginaali.
6. Tehdään veloituksesta sellainen, että käyttäjän kannattaa ilmoittaa paras tuntemansa arvio liikenteen keskiarvosta. Tällä tavalla itse ongelma siirretään käyttäjille, mikä saattaa olla monesti kyseenalaista: kaikilta asiakkailta ei voida edellyttää asian vaatimaa ymmärrystä liikenteen ominaisuuksista.

Tilaajaverkko-osuuden aiheuttamia ongelmia

Kuvan 3 mukaisesti julkisen tai muun suuren verkon operaattorin eräs keskeinen liikenteen hallinnan ongelma on varsinaisen käyttäjän ja julkisen verkon välinen verkko-osuus, johon kummallakaan ei ole juurikaan vaikutusmahdollisuutta. Käyttäjä tai pikemminkin joku sovellus, joka liikennettä tuottaa, voi lähettää vakionopeuksista liikennettä, mutta kuljettuaan tilaajaverkon

läpi (I3:sta I2:een) sen ominaisuudet voivat muuttua huomattavasti. Julkisen verkon kannalta tämä on sangen hankalaa, koska yhteyden ominaisuudet tunnetaan tällaisessa tapauksessa huonosti, jolloin esimerkiksi puskurikapasiteetin riittävydestä verkon sisällä ei voida olla varmoja.

Ratkaisumalleja:

1. Varsin suoraviivainen ratkaisumalli on liikenneprosessin muokkaaminen liityntärajapinnassa mahdollisimman tasaiseksi. Tämä ratkaisu sopii erityisen hyvin vakionopeuksisille (tai yleisemmin on/off-tyyppisille) yhteyksille, joilla oikea nopeus tunnetaan. Sensijaan vaihtelevanopeuksisille yhteyksille tasoitus on käytännössä suoritettava huippunopeuteen, eli liikenneprosessia ei yleensä voida palauttaa alkuperäiseen muotoonsa. Oleellisin ongelma, eli tulevien solujen klimppiintyminen voidaan tällä menetelmällä kuitenkin ratkaista.
2. Valitaan yhteyttä määrittelevät parametrit siten, että normaalilla tilaajaverkon liikennekuormituksella parametrien arvojen ylittäminen on hyvin epätodennäköistä. Tähän tarkoitukseen käytetään erityisesti CDVT-parametriä.

Tästä aiheesta on julkaistu lukuisia tutkimuksia, mutta ongelmaksi jää aina se, että tilaajaverkko-osuuden hetkellinen kuormitus ja liikenneprosessin luonne vaikuttaa olennaisesti syntyvään viiveeseen ja sitä kautta myös siihen, mikä on järkevä arvo CDVT:lle. Toisaalta jos CDVT:n arvo on turhan suuri, niin julkisen verkon yhteys saattaa jäädä saamatta, koska puskurikapasiteetin riittävydestä ei voida olla varmoja.

3. Veloituksella voi olla myös oma vaikutuksensa: olkoon CDVT:n vaikutus yhteyden hintaan oleellinen. Tällöin asiakkaan voi olla kannattavaa hankkia omat liikenteen muokkaimensa, jolloin julkisen verkon kannalta ongelma saadaan poistettua.

Häiriköt

Useimmat atm-verkon liikenteen hallinnan ongelmat olisivat kohtuullisen helposti ratkaistavissa (tai niitä ei ilmenisi lainkaan), jos voitaisiin olettaa, että kaikki asiakkaat käyttäytyisivät verkon toiminnan kannalta mahdollisimman järkevästi. Julkisessa verkossa, jossa jokainen asiakas joutuu maksamaan saamastaan palvelustaan, ja jossa asiakkailla ei ole mitään työ- tai muuta vastaava suhdetta verkon omistavaan yhtiöön, näin ei kuitenkaan voida olettaa.

Vähintään on syytä olettaa, että osa asiakkaista haluaa lähes keinolla millä hyvänsä minimoida tiedon siirrosta maksamansa hinnan. Varsinkin jos veloitusjärjestelmä on monimutkainen, tähän saattaa olla hyvät mahdollisuudet. Veloitusjärjestelmän suomien mahdollisuuksien hyväksikäyttäminen ei sinänsä ole kiellettyä, vaikkakin joissakin tapauksissa hyväksyttävyydestä voidaan ainakin keskustella. Esimerkiksi TCP/IP -protokolla toimii tiettyjen periaatteiden mukaan, jotka on määritelty jossain suosituksessa. Asiakas saattaa kuitenkin pyrkiä muuttamalla kyseisen protokollan parametrejä hyötymään muiden asiakkaiden kustannuksella.

Toinen ongelma liittyy liikenteen tilastolliseen luonteeseen. Tilastollisia tarkasteluja varten (esim. yhteyttä hyväksyttäessä) on lähes pakko tehdä tiettyjä oletuksia, kuten että eri yhteydet ovat tilastollisessa mielessä keskenään toisistaan riippumattomia. Jos tilastolliset oletukset eivät pidä paikkaansa, suoritettavat laskelmat voivat johtaa epätoivottavaan lopputulokseen.

Ongelmaksi asia muodostuu, jos asiakas operaattoria harhauttamalla saa itselleen merkittävää etua. Näin käy vaikkapa on/off-lähteen tapauksessa, jos huippunopeus on suuri (esim. 10% väylän

kapasiteetista) ja on-tilan todennäköisyys pieni (esim. 5%). Tällöin tilastollista kanavoinnista saatava hyöty jää todellisuudessa hyvin pieneksi ja vastaavasti käyttäjältä perittävä veloitus on korkea. Itse asiassa useimmat CAC-menetelmät eivät näin suurella huippunopeudella hyödynnä tilastollista kanavointia lainkaan vaan hyväksytyjen yhteyksien määrä on korkeintaan 10. Teoriassa vielä 20 yhteydellä solujen menetystodennäköisyys on alle 10^{-9} . Sensijaan, jos käyttäjä muodostaa 10 kappaletta yhteyksiä, joilla huippunopeus on 1% väylän kapasiteetista ja on-todennäköisyys 10%, tilanne on oleellisesti toinen. Tyypillinen CAC hyväksyy noin 1000 yhteyttä ja teoriassa jopa 1132 mahtuu. Eli sallittu kuormitus saattaa käytännössä olla 5 - 10 kertainen! Vastaavasti 10 hitaan yhteyden hinta voi olla olennaisesti alhaisempi kuin yhden nopean.

Tämän kaltaisiin ongelmiin voisi ajatella seuraavia ratkaisumalleja:

1. Pyritään niin tiukkaan liikenteen valvontaan, että mikään muodollisesti oikea liikenneprosessi ei aiheuta verkossa ongelmia. Samalla tilastollisen kanavoinnin hyödyntämisen mahdollisuudet heikkenevät oleellisesti, koska tilastollisia suureita ja erityisesti eri yhteyksien välisiä riippuvuuksia on erittäin vaikea valvoa.
2. Pyritään muotoilemaan asiakkaan kanssa tehtävä sopimus siten, että asiakkaan ei kannata yrittää harhauttaa operaattoria. Eli kielletään esimerkiksi yhden yhteyden pilkkominen moneen osaan "hyötymistarkoituksessa" riittävän suuren rangaistuksen uhalla. Tosin tätä sääntöä saattaa olla hyvin vaikea valvoa, muutenhan siitä saatava hyöty jää vähäiseksi.
3. Laaditaan veloituseriaatteet siten, että harhauttamisesta ei ole ainakaan olennaista etua asiakkaalle. Tähän suuntaan voidaan vaikuttaa esimerkiksi jokaisen yhteyden muodostamiseen liittyvällä maksulla. Toisaalta veloituksesta ei saa tehdä liian monimutkaista.
4. Hyödynnetään reaaliaikaisia liikennemittauksia. Tällöin täytyy kuitenkin päästä käsiksi *olennaisiin* parametreihin eli solujen menetyksiin ja viiveisiin, jotka taas ovat yleensä mitattavissa ainoastaan koko liikenteelle. Vaikeus piilee siinä, että mahdollisten ongelmien aiheuttajaa voi olla hyvin vaikea paikallistaa, joten "takaisinkytkentä" saattaa jäädä puutteelliseksi, vaikka palvelun taso voitaisiinkin turvata. Poikkeukselliseksi katsottava käyttäytyminen lienee kuitenkin mahdollista havaita.

5. Traffic control and congestion control

5.1 The challenge of traffic control

The principle of ATM itself guarantees neither high utilisation nor high Quality of Service without traffic control. Congestion in its various forms is the basic problem of traffic control in the ordinary telephone network, in packet network as well as in the ATM network. Congestion occurs when the demand is greater than the available resources. According to Jain (1990) congestion is caused in packet networks:

- by a shortage of a buffer space,
- by slow links or

- by slow processors, and

this may lead to a belief that, when some or all of these problems are solved by technical development (cheap memory, high speed links and processors), the congestion problem goes away. Contrary to this belief, without proper protocol redesign, technical development may lead to more congestion and thus reduce performance (Jain 1990). This is indisputably the situation in ATM networks as well, and all over the world there is a vast effort to develop proper control methods for ATM. To quote Gilbert, Aboul-Magd and Phung (1991): the challenge is to design simple and efficient controls while still achieving reasonable bandwidth utilisation through a statistical multiplexing.

Recently there have been some proposals for complicated control architecture. The basic idea in those proposals is that several classes of traffic with different QoS requirements are considered explicitly at every level of system design, both at the edge and at the core of the network. Therefore the network should be able to allocate the buffer capacity according to the actual requirement of each connection, not on the First in First out (FIFO) basis as in traditional control scheme of ATM networks.

This study follows the main line in standardisation and supposes that the separation of different services, if applied, is done by higher protocol layers and no parallel buffers at the core of ATM network are used (with the possible exception of separate buffers for high and low priority flows).

5.2 Definitions

This section depicts the role of traffic control and congestion control as they have been defined in recommendation I.371 of International Telecommunication Union, Telecommunication Standardization Sector (ITU-T 1993a). The *primary* role of traffic control and congestion control is to protect the network and the user in order to achieve network performance objectives. An *additional* role is to optimise the use of network resources.

Traffic control refers to the set of actions taken by the network to avoid congested conditions. Congestion control refers to the actions taken by the network to minimise the intensity, spread and duration of congestion. Congestion is defined as a state of network elements in which the network is not able to meet the network performance objectives. It is to be distinguished from the state where buffer overflow is causing cell losses, but still meets the negotiated Quality of Service.

Traffic control functions are (ITU-T 1993a):

- Network Resource Management (NRM): Allocation of network resources in order to separate traffic flows according to service characteristics.
- Connection Admission Control (CAC): A set of actions taken by the network during the call set up phase in order to establish whether a virtual channel (or path) request can be accepted or rejected.
- Usage/Network Parameter Control (UPC/NPC): A set of actions taken by the network to monitor and control traffic, in terms of traffic offered and validity of the ATM connection. The main purpose of UPC is to protect network resources from malicious as well as unintentional misuse.
- Priority control: the user may generate different priority traffic flows by using the Cell Loss Priority bit. A congested network element may selectively discard cells with low priority if necessary.

- Traffic shaping is a mechanism that alters the traffic characteristics of a stream of cells to achieve a desired modification of those characteristics. Examples of traffic shaping are peak cell rate reduction and burst length limiting.
- Fast Resource Management (FRM): A typical FRM function allows a network to allocate capacity for the duration of a burst in response to a user request.

Congestion control functions are:

- Selective cell discarding: A congested network element may selectively discard cells identified as belonging to a non-compliant ATM connection and/or those cells with lower Cell Loss Priority.
- Explicit Forward Congestion Indication (EFCI) may be used to assist the network in avoidance of and recovery from a congested state. A network element in a congested state may set an Explicit Forward Congestion Indication in the cell header so that this indication may be examined by the destination customer equipment.

As regards NRM, the service separation has an important effect on CAC because the multiplexing process is more regular if the traffic characteristics of aggregated streams, such as peak rate and burst length, resemble each other. Furthermore, it might be possible to use simpler CAC methods if sources are grouped into few service classes. These service classes may have various cell loss requirements in which case the network utilisation can be improved provided that different services use different links. However, the main reason to introduce multiple QoS classes is that they can be used to protect higher priority flows against cell loss during periods of short term traffic overflow.

According to Eckberg, Lucantoni and Prasanna (1991) there are two issues that must be addressed with respect to the Cell Loss Priority (CLP) indicator:

- the QoS given to the CLP=0 [higher priority] traffic must be only insignificantly affected by the CLP=1 traffic and
- some utility from the CLP=1 traffic must be derivable by the end-terminals.

In the first condition the analysis of CLP=0 traffic, which is the chief concern of this study, is almost independent of the CLP=1 traffic flow. Similar procedures that are used with CLP=0 traffic can be applied for the CLP=1 traffic (or for the combined traffic) using different QoS parameters.

The principle of ATM makes it possible to change the traffic stream before multiplexing mainly in order to increase the utilisation of network links, in particular when the burstiness of offered traffic is very high. The usefulness of this approach depends on the time-scale of variations and on the delay requirements of application. As regards the performance evaluation the effect of traffic shaping can be included in the offered traffic process.

The idea of FRM is to increase the multiplexing efficiency by implementing admission control at the burst scale (or at the rate-variation scale) in addition to the connection scale. When a new burst is to be sent it is necessary to obtain a new resource allocation by means of a rapid in-band signalling exchange between user and successive network nodes. The bandwidth used by a connection is relinquished at the end of a burst. According to Roberts the limitations of this approach are the time needed to obtain a new resource allocation which reduces efficiency particularly for short bursts, the need to implement a sophisticated protocol and the low network utilisation realisable when the connection peak rate is high. If a network node rejects the burst, it can be either buffered at the network interface or discarded depending on the application. Buffering is unavoidable if the application is file transfer whereas with real time applications, such as voice and video, it is not sensible to buffer bursts for re-transmission. In both cases the rejecting probability should be reasonably low to avoid enormous buffers or degradation of QoS.

There are two Fast Reservation Protocols (FRP) for the realisation of FRM:

- *Delayed Transmission* (FRP/DT) is based on the prior negotiation and reservation of a peak rate value on each switching node along the connection using special management cells;
- *Immediate Transmission* (FRP/IT) supposes that link capacity can be reserved "on the fly" by the first cells of a burst when it arrives in each switching node and on each link of its path.

In the case of blocking, special procedures would be necessary to inform the user to allow him to make a new attempt.

Congestion indication can be send backwards by using Backward Explicit Congestion Notification (BECN). When a queue in an ATM switch exceeds a certain threshold it sends BECN cells back to the sources of virtual channels currently submitting traffic to it. On receipt of a BECN cell to a particular virtual channel, a source must reduce its transmission rate for the indicated channel. If no BECN cells are received for a certain period of time, a source may gradually restore its transmission rate. According to Newman BECN could be applicable for high-burst sources without specifying traffic characteristic for every individual data source when the transmission delay is limited, as in LAN, but considerable problems might arise if the network's size is large (a good performance level might be extended to a transmission delay of several hundred kilometres).

5.3 Preventive vs. reactive control

There are two basic approaches for controlling broadband networks: preventive and reactive. The preventive approach relies mainly on traffic control functions while the reactive approach utilises primarily congestion control functions. The basic idea of reactive or feedback control is that the network allows the offered traffic increase until the capacity of a link is exceeded or, in a more advanced case, until some network element detects that an overload situation is probable.

To quote Blaabjerg (1991): In Europe the trend has been towards a simple and preventive strategy, based on the ideas from traditional telecommunications community whereas in the US a trend towards a more dynamic strategy based on ideas from computer communication community is seen. A good compromise, as Ramamurthy and Dighe have proposed, would be an aggressive congestion avoidance strategy that uses network resources optimally, with reactive control mechanisms as backups to relieve congestion in the unlikely event of the network experiencing congestion.

Feedback control has been proved to be useful in data networks where sources are suitable for cell rate re-allocation, buffers in network nodes are typically large and bit rates are not very high. Unfortunately, the situation is almost the reverse in a typical ATM network because it will be very hard to re-allocate most sources, buffers are small and bit rates are very high.

ATM networks, particularly in large areas, are dependent on the capabilities of preventive control methods, but feedback control functions can still be useful in minimising the intensity and duration of congestion. In addition, reactive functions may have an important role when exploiting the free capacity in ATM networks. Because of the statistical properties of traffic in ATM networks the mean load of high priority traffic may remain low, even less than 0.1. Network operators may attempt to utilise the remaining capacity by offering it to customers who have a large amount of data to be transferred but can tolerate occasional long delays. In addition, traffic sources should be able to reduce the bit rate because even a large buffer will overflow quite soon if the link is fully reserved by high priority flows.

5.4 Response times

The response time defines how quickly the controls react. Figure 5.1 shows a typical classification of control functions according to the response time. A clear similarity can be seen between Figure 1.2 (Time resolution of ATM traffic process) and Figure 5.1: the *connection scale* and *cell scale* in Figure 1.2 correspond respectively to *connection duration* and *cell time* in Figure 5.1.

There is an obvious relation between *rate-variation* and *burst scales* in Figure 1.2 and *round trip propagation time* in Figure 5.1 although those time levels are based on two different phenomena, the properties of traffic offered and the properties of the network. The round-trip delay in a wide area ATM-network may vary from 1 ms to 100 ms. This is the typical time scale of the arrival process of bursts and it also partly covers the rate-variation scale fluctuations. These similarities in time scales are important but they do not entirely explain the complicated relationship between offered traffic and feedback control functions (FRM, EFCI, BECN); we should take into account many other aspects, such as delay and delay variation requirements, upper layer protocols, and limited buffer capacity.

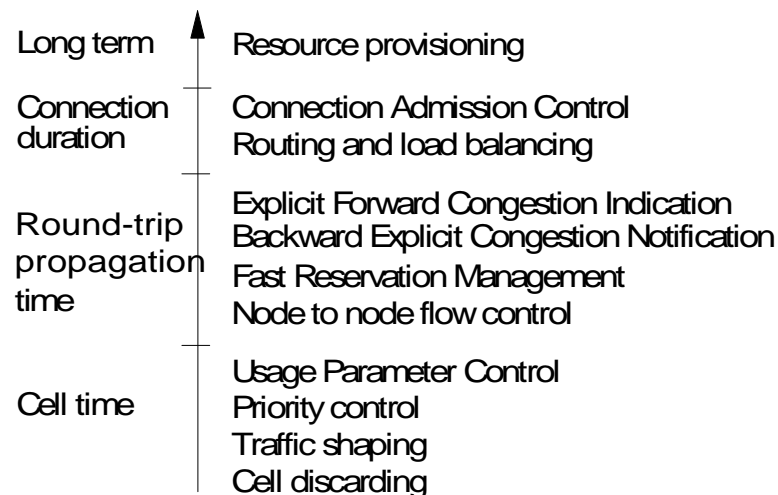


Figure 5.1. Control response times (ITU-T 1993a).

6. A proposal: Traffic management tools for ATM networks with real-time and non-real-time services

6.1 Introduction

One of the inherent properties of ATM networks is the diversity of services. Firstly, the mean and peak cell rates of connections may vary enormously based on the application used by the customers. Secondly, customers have very different Quality of Service (QoS) requirements. A network operator has a difficult task to meet all the differing requirements and keep the network utilization high at the same time. Let us take a simple example where the link capacity is 130 Mbit/s and there are 6 on/off sources with peak rate 25 Mbit/s and mean rate 5 Mbit/s (Fig. 6.1). Evidently, the overload probability in this case is so high that the QoS will be somehow deteriorated. One of the main tasks of traffic management in ATM networks is to solve this fundamental problem.

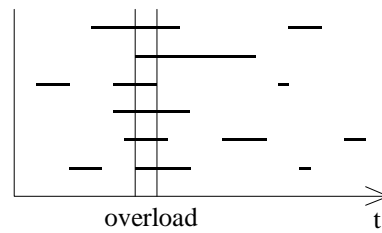


Fig. 6.1. Traffic demand of 6 sources and an overload situation.

The solutions can be divided into the following groups:

1. *Traffic shaping.* In this approach all (or at least almost all) traffic variations are smoothed before the ATM network. The disadvantage of this approach is apparent: there must be huge buffers at customer interfaces, because the time scale of traffic variations could be several seconds or even minutes and peak rate could be as high 25 Mbit/s as in the example. These buffers are expensive and may bring about an unacceptable long delay and delay variation.
2. *Admission procedure for each connection.* It is possible to avoid overflow situation by rejecting each connection request that endangers the QoS of other connections. In the above case, this means that at most 5 connections (or perhaps 4) will be accepted on the link. Some network operators may consider the average load obtained with this approach far too low. However, with certain applications this scheme might be practical, or even unavoidable.
3. *Admission procedure for each information block.* It is possible to do the capacity checking separately for each information block. This protocol is called ATM block transfer capability (ABT) [1]. There are two different variants of this service: ABT with delayed transmission and ABT with immediate transmission. In the first approach, each information block will be sent to the network without any preceding acceptance procedure, and in the second one there is a CAC type of procedure before the sending of each information block. With both approaches, a sufficient, and probably quite large, buffer capacity is needed at the user interface because of the relatively high probability that there is not enough capacity in the network just when the user wants to transmit the information block. A large buffer means always possibility for long delay.
4. *No admission control and small buffers.* In this approach all connections will be accepted in the network and due to small buffers high cell loss ratios are unavoidable. The advantages of this approach are high utilization and short delay. But at the same time, because of the high cell loss ratio the benefit obtained by customers could be small, or even nonexistent, during an overload situation. One method to improve the situation is to classify cells as important and less important cells.
5. *No admission control and very large buffers.* In the previous approaches it is possible to use relatively small buffers at each network node. We can also use very large buffers in order to store all cells that cannot be immediately transmitted forward. However, because of the high speed of ATM-networks and the properties of data traffic this buffer should be really large if we want to avoid cell losses and at the same time keep the utilization high. For instance, if in our simple example the overload situation lasts only 1 s, the needed buffer capacity is almost 50 000 cells.

Furthermore, if we have huge buffers there will also be very long delays and large delay variation.

6. *No admission control, large buffers and intelligent scheduling.* Because the buffer capacity will in practice be restricted there should be rules for the acceptance and rejection of cells during overload situation. This selection should be efficient on the network point of view and, above all, fair for all users. In our example, all users will lose some cells when the buffer occupation exceeds certain limit. It is the responsibility of each user to react properly on the increased cell loss ratio.
7. *Reactive control.* All the above approaches are based either on preventive control or on the assumption that users can react properly on the impaired QoS caused by a network overload. A further approach is to construct detailed protocols for regulating the traffic process sent to the network, as in the Available Bit Rate (ABR) service [1]. It should be noted that this approach requires large buffers both at the network nodes (because of high bit rates and round-trip delays), and at user interfaces since we can hardly suppose that all applications can immediately adapt to any changes of available bit rate. The main advantage of this approach is that it might be possible to keep the cell loss ratio low and utilization high, but again, at the expense of long delays.

What conclusions can we draw from the above general considering? First of all, Fig. 6.1 shows that it is impossible to attain at the same time small cell loss ratio, short delay and high network utilization for all connections if there is a considerable amount of data traffic with high burstiness and high peak rate. This conclusion can be illustrated by a tetrahedron shown in Fig. 6.2.

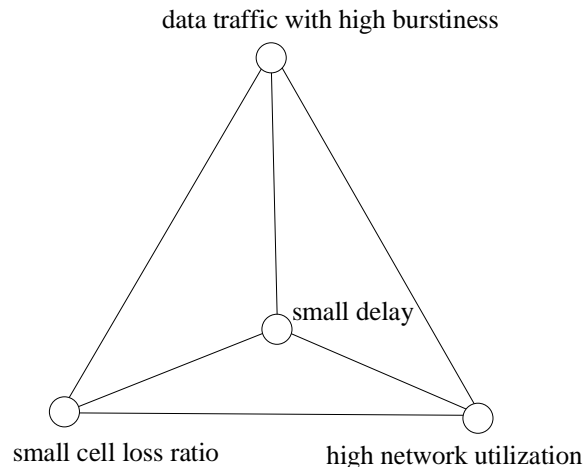


Fig. 6.2. Four targets of ATM services [2].

By adjusting the above 7 service items in the framework of Fig. 6.2 we can find 4 basic services:

1. Constant bit rate service with small cell loss ratio, small delay, high utilization, but without the possibility to transmit bursty data traffic as such (item 1).
2. Variable bit rate service for bursty data traffic with small cell loss ratio and small delay, but without high network utilization (item 2).
3. Unspecified bit rate service with small delay and high utilization but without any hope for small cell loss ratio (item 4).

4. A network service with small cell loss ratio and high utilization, but long delay for the application during overload situation. In item 3 the delay arises at user interface (because all information blocks cannot be transmitted immediately) and in items 5, 6 and 7 mainly at network nodes but also at user interfaces.

In consequence, the network operator must abandon either small cell loss ratio, small delay, high utilization or bursty data traffic with high bit rate. In other words, any network service forms a plane which goes through 3 edges of the tetrahedron but misses one of the edges. Of course, it is always possible to offer several network services at the same time and meet all the four targets for different connections: small delay and small cell loss probability for CBR connections and high utilization with UBR (Unspecified Bit Rate) type of service offered to data traffic. One of the main questions of the network operator is what is the best selection of the ATM services to be offered to customers. The rest of this paper is dealing with this question.

6.2 Service Combination

Internet and its explosively increasing traffic demand is clearly one of the most important elements as regards the near future of ATM networks. The applications running in the current Internet have to adapt to the service characterized by long delay and high frame (or packet) loss ratio. It is assumed that a higher protocol level can adapt to this, somewhat poor, QoS by re-sending the lost frames and by decreasing the bit rate after frame losses. It is very probable that this service model will prevail several years although the ATM technology will give new opportunities and a lot of additional capacity for Internet. Therefore, in an ATM network there has to be a service which meets this actual service demand. As to the above presented service items the most suitable service model for this demand is item 6: no admission control, large buffers and intelligent scheduling. This approach will be addressed more thoroughly later in this paper.

However, as Scott Shenker has stated in his article [3], this UBR service is not sufficient but, in addition, an ATM network operator should offer a service which satisfies the requirements of real-time applications like voice and video. Many real-time applications are intrinsically variable bit rate sources. The basic problem is how we can exploit the statistical multiplexing of VBR connections when we have a very stringent requirement for cell loss ratio and we only know the traffic process approximately. This is a hard problem but after extensive studies there are at least promising solutions (although the final assessment cannot be made until we have experience in real networks). One possible scheme is presented in the following chapter.

Thus, our suggestion is that a basic ATM node should have two service types: a service for real-time CBR and VBR traffic with high QoS requirements, and an inexpensive UBR service for applications capable to adapt into changing capacity and poor QoS. Although there certainly are applications not very suitable for these two service types, most of the current user demand can probably be satisfied with these services. The main advantage of this approach is that the integration of the two service types in an ATM node is quite straightforward. The only requirement is that the node must separate the services by a strict priority: if there is any cell in the buffer belonging to the real-time service class, UBR cells are not transmitted forward. As for the traffic management of real-time services, the ATM node looks (almost) like a simple switch with only one service class. The only complication emerges if the operator wants to guarantee a minimum cell rate even for some UBR connections; this cell rate should be taken into account in the CAC procedure of the real-time services.

6.3 Realization of Real-Time Service for CBR and VBR Connections

The basic requirements for the realization of a real-time service are small delay, (usually) small cell loss ratio, and possibility to exploit statistical multiplexing. Because of the small delay requirement preventive control approach is the most feasible one. In order to realize this scheme the following traffic management components are needed: traffic descriptor for the characterization of each connection, usage parameter control (UPC) for controlling the traffic sent to the network by each user, and connection admission control (CAC) to decide whether a new connection can be accepted into the network.

The properties of a connection are characterized by four traffic parameters: peak cell rate (PCR), sustainable cell rate (SCR), cell delay variance tolerance (CDVT) and intrinsic burst tolerance (IBT) [1]. These parameters are then controlled by UPC which accepts certain traffic patterns and rejects some others. Although it is not quite clear on the network performance point of view which one of the acceptable traffic patterns is absolutely the worst one, the pattern presented in Fig. 6.3 is at least near the worst and due to the simplicity suitable for practical performance evaluation.

Traffic parameters and UPC are well defined but what has been missing is a simple and efficient CAC that guarantees high QoS. The fundamental problem of this task is that the aggregate traffic process is very difficult to model with any simple formula. The solution presented in [4] is to firstly divide the problem into three time scales: cell, burst and rate-variation scale.

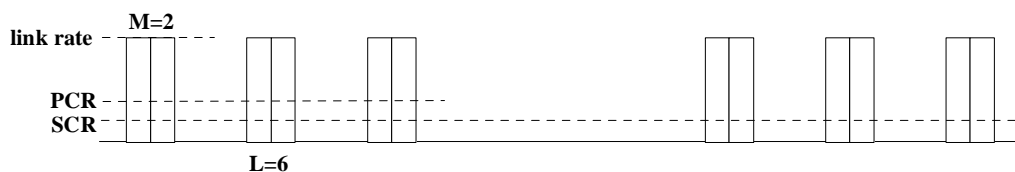


Fig. 6.3. A near worst case traffic pattern with given PCR, SCR, CDVT and IBT; M = the maximum number of consecutive cells; L = the maximum burst size in cells.

The cell scale process is composed of the short term fluctuations in the traffic process. As to the (near) worst case traffic, cell scale takes into account the maximum number of consecutive cells (M). Now we can neglect temporarily the other traffic parameters and attempt to calculate the allowed load (denoted by ρ_{cell}) taking into account only this traffic process. A feasible approach is to apply an $N^*D/D/1/K^*$ model where the real buffer capacity (K , in cells) is replaced by $K^* = K/M$. This approach offers a good basis for practical calculations, and for the use of pre-calculated tables. The superposition of cell scale variations can be modeled by using the effective bandwidth method.

The other extreme case is the rate-variation scale where the traffic fluctuations are slow. In practice this assumption is valid always when the maximum number of cells in a burst (L) is larger than the buffer capacity (see [4]). In this case we can ignore the effect of buffer capacity and consider only peak and sustainable cell rates (e.g., [5]). Large deviation approximation is very useful when evaluating homogeneous cases, and effective variance method gives good approximation for the superposition of different sources, see e.g. [6] and [7].

The most difficult area exists between the cell and rate-variation scales. Unfortunately, it is very difficult to solve the allowed number of connections with the exact deterministic process presented in Fig. 6.3. One simpler approach is to replace the deterministic process by a Markovian one with the same mean and peak rates and to neglect the cell clumping due to cell delay variation. This Markovian model is solvable although the solution needs a lot of numerical processing. Fortunately, our recent studies indicate that it is possible to develop a simple approximation for this burst scale process.

The remaining problem is to combine the results of the three scales. Our solution is a simple combination of effective bandwidth and effective variance models where a new connection is accepted if the following condition is valid [4]:

$$\sum_i m_i + \sqrt{\left(\sum_i \sigma_i^*\right)^2 + \sum_i v_i^*} \leq \rho_{\max} C \quad (1)$$

where: m_i = sustainable rate, C = link capacity,

$$\sigma_i^* = \sigma_{i,cell}^* + \sigma_{i,burst}^*,$$

$$\sigma_{i,cell}^* = m_i \left(\frac{1}{\rho_{i,cell}} - 1 \right),$$

$$v = v_{i,burst}^* \quad \text{or} \quad v = v_{i,rsv}^* \quad \text{depending mostly on the maximum burst size to buffer size ratio, and}$$

$$v_{i,rsv}^* = SCR(PCR - SCR) \left(c_1 + \frac{c_2}{e^{c_3 SCR/PCR} + (PCR/C)^{c_4} + c_5} \right).$$

The tuning of coefficients c_1 , c_2 , c_3 , c_4 and c_5 may be done by using a genetic optimization method [7]. The approximate formulae for burst scale parameters ($\sigma_{i,burst}^*$ and $v_{i,burst}^*$) are still under study but interim simulation studies have shown that an appropriate accuracy is obtainable. In consequence, it is possible to develop a relatively simple CAC method that takes into account the four traffic parameters, all time scales and gives good possibility for statistical multiplexing.

Our performance evaluation has covered the following parameter region: $PCR/C = 0.0002 \dots 0.25$, $SCR/C = 0.0001 \dots 0.20$, $SCR/PCR = 0.001 \dots 1$, $M = 1 \dots 20$, $L/K = 0.01 \dots 3$. The most difficult parameter combination is when $PCR/C > 0.1$ and $SCR/PCR < 0.1$. Then the real cell loss ratio depends in a complicated way on the parameters of other connections and even (1) fails in some cases to catch the real behavior of the aggregate traffic process. However, the overall results presented in Table 6.1 are very promising. As the total number of cells in the simulation is 16 400 000 000, the results provide a good assessment of the performance of the CAC algorithm. Yet, it should be noted that the worst case represents 22% of the total amount of lost cells, and there may be even worse cases not occurred in our simulations.

Table 6.1. Results of performance evaluation of CAC algorithm, the target cell loss ratio = 10^{-5} , $K = 100$, in each case 5 000 000 time slots are simulated using one randomly selected source type and CBR load (0 ... 99%).

selection of cases	number of cases	number of lost cells	average cell loss ratio
$P_{loss} > 2 \cdot 10^{-4}$	1	1939	$3.95 \cdot 10^{-4}$
$2 \cdot 10^{-4} > P_{loss} > 10^{-5}$	41	6028	$4.83 \cdot 10^{-5}$
$10^{-5} > P_{loss} > 2 \cdot 10^{-7}$	116	933	$2.50 \cdot 10^{-6}$
$P_{loss} = 0$	4582	0	0
all	4740	8900	$5.42 \cdot 10^{-7}$

6.4 Realization of a Fair UBR Service

The main aspect of UBR service is fairness. In order to provide bandwidth fairness among competing UBR connections, the buffer capacity at each switch needs to be allocated fairly among the competing connections. Otherwise a connection that gets more than its fair share of the buffer space, also gets more than its fair share of the bandwidth. The allocation scheme presented in this paper allows a fair share of link capacity using only a pure FIFO buffer [8].

The basic idea of the proposed scheme is that the buffer implementation should be as simple as possible, whereas we may allow a relatively complex algorithm to decide whether an incoming cell should be accepted or rejected. Let us suppose that the incoming cells to an ATM node are generated by several sources, all of which send AAL5 frames (then we know when a frame ends). Firstly, it is important to know whether the arriving cell is the first cell of a frame, and whether some cells of the same frame have already been delivered forward. If there is an impending danger of buffer overflow, whole frames should be dropped instead of individual cells. Therefore, if the buffer occupancy exceeds a certain limit, the first cell and all the following cells of a frame should be dropped. Consequently, if the first cell of a frame is accepted into the buffer, all the following cells of the frame will be accepted, provided that the buffer is not fully occupied.

Now the first algorithm (AI) can be defined as follows:

- The first cell of an AAL5-frame is dropped if $X > R$, where X is the number of cells in the buffer and R is a limit for buffer occupancy.

Unfortunately, this buffer allocation scheme does not guarantee a fair share between different type of connections if some connections have exploited more resources than the other ones. If the network operator's intention in an overload situation is to share the link capacity evenly among all active connections, then the operator should drop cells from those connections that have exploited the largest part of the link and buffer capacity. One way to do this is to consider the number of cells in the buffer. Let us denote the number of cells connection i has in the buffer by Y_i and the number of connections which have at least one cell in the buffer by N_a . If during an overload situation a connection has more cells in the buffer than the average value (i.e., $Y_i > X/N_a$), we may conclude that the connection is to some extent responsible for the overload situation. Let us denote

$$W_i = \frac{Y_i N_a}{X}. \quad (2)$$

Parameter W_i can be used as a measure of the exploitation of network resources by connection i provided that the buffer capacity is sufficiently large in comparison with the typical burst size. Now we can define an advanced algorithm by applying parameter W_i . In this algorithm (A2) the first cell of an AAL5-frame is dropped if

$$(X > R) \text{ and } (W_i > 1).$$

Although this simple scheme levels down in some degree the differences in link capacity used by connections, the result is not quite satisfactory. When X exceeds R all connections which have more cells in the buffer than average (i.e., $W_i > 1$) experience roughly the same frame loss ratio, independently of the instantaneous rate of the connection.

In order to improve the performance of the algorithm we can replace the on/off type rejection function by a smoother one. In the third algorithm (A3) the first cell of an AAL5-frame will be dropped if:

$$(X > R) \text{ and } W_i > Z \left(1 + \frac{K - X}{X - R} \right), \quad (3)$$

where K is the buffer capacity in cells and Z is a free parameter (typically less than 1).

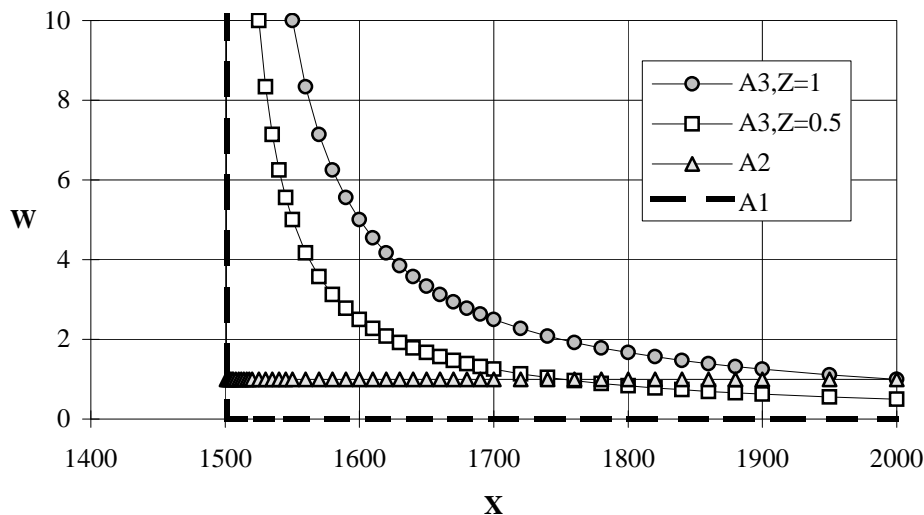


Fig. 6.4. Rejection functions of Algorithms 1, 2 and 3; $K = 2000$, $R = 1500$.

The shapes of the rejection functions of different algorithms are presented in Fig. 6.4. If we apply Algorithm 3 and the buffer occupancy exceeds R , cells are rejected from a connection only if it has a considerably amount of cells in the buffer. When X approaches K , the allowed number of cells may decrease eventually below 1. As a consequence, finally most of the first cells of AAL5-frames will be dropped, because the scheduling algorithm tends to level down the differences in the number of cells while X is increasing. This property is desirable since some of the buffer capacity should be left for the remaining cells of accepted frames. An appropriate behavior of algorithm can be obtained by a proper selection of parameters R and Z .

The results presented in [8] have shown that the proposed algorithm makes it possible to offer high fairness with a simple queuing principle. Especially important is that the connections with moderate bit rate may remain totally undisturbed as regards the cell losses although the delay variation is similar to all connections. If the higher layer protocols work properly, the overload situation finishes after some delay and the highest bit rates are cut to a certain value.

6.5 Conclusion

The goal of this paper has been to show that it is possible to meet most of the real user demand by a simple ATM switch with two service classes: a real-time service class for CBR and VBR connections and UBR service class for applications with loose QoS requirements. The main advantage of this simple approach is that the integration of the two service types in an ATM node can be realized using a simple priority scheme.

Real-time services need a feasible preventive control method. This control method can be composed of three parts. Two parts are well defined in ITU's specifications, namely, traffic descriptor and usage parameter control. This paper presents the outlines of the third part, Connection Admission Control. Our investigations, although yet in preliminary stage, show that our framework forms a good basis for practical implementation with good performance.

The other necessary part of traffic control is a fair scheduling algorithm for UBR service. This paper shows that it is possible to attain high fairness by using FIFO buffers and a simple allocation scheme. The advantage of the proposed allocation algorithm is that it reacts accurately and quickly during overload situations and reduces the bandwidth of only those connections that are using an excessive amount of the link bandwidth. At the same time, connections with small cell rate do not experience any cell losses. By using weighting coefficients it is possible to allocate link capacity according to pre-defined values. However, further simulations with several nodes and upper protocol levels are needed to assess the real performance of the buffer allocation schemes.

References

- [1] ITU-T, Recommendation I.371, Traffic Control and Congestion Control in B-ISDN, Geneva, May 1996.
 - [2] J. Virtamo, Inauguration speech, 18.9.1995, Otaniemi, Espoo, Finland.
 - [3] S. Shenker, Fundamental Design Issues for the Future Internet, IEEE J. on Selected Areas in Communications, Vol. 13, No. 7, Sept. 1995.
 - [4] K. Kilkki, Traffic Characterisation and Connection Admission Control in ATM networks, Doctoral thesis, Espoo 1994.
 - [5] J. Roberts (ed.), Performance Evaluation and Design of Multiservice Networks, COST 224 Final Report, Luxembourg, 1992.
 - [6] K. Kilkki, Connection Admission Control Methods in ATM networks, International Teletraffic Seminar, Bangkok, 1995.
 - [7] S. Saranka, K. Kilkki, Optimization of Effective Variance Based CAC Algorithms, to be appeared in Globecom'96, London, 1996.
 - [8] J. Heinänen, K. Kilkki, A Fair Buffer Allocation Scheme, submitted for publication, 1995.
-