The performance of TCP/IP over an operational ATM network

van den Bergh, M.M.G.N.

Award date:
1995

Technische Universiteit Eindhoven

Faculty of Electrical Engineering
Section of Digital Information Systems

Master's Thesis:

The performance of TCP/IP over an operational ATM network

M.M.G.N. van den Bergh

Coach:	ir. F.C.I. van den Eijnden, ir. P.H.A. Venemans (KPN Research)
Supervisor:	Prof.ir. F. van den Dool
August 1995	

The Faculty of Electrical Engineering of Eindhoven University of Technology does not accept any responsibility regarding the contents of Master's Theses.

KPN Research

Communication Architectures and Open Systems

Information sheet issued with Report R&D-SV-95-692

Title: The performance of TCP/IP over an operational ATM network

Abstract: One of the first opportunities for network operators to exploit the ATM network commercially, is to offer TCP/IP connections over ATM in the near future. In this report the interworking between TCP/IP and ATM is studied, performing a number of ATM network experiments. Experience with the available ATM equipment and the traffic control functions is gained, while the influence of several settings and parameters on the performance of TCP/IP over ATM is examined. Furthermore, the distinction between the Network Performance and Quality of Service is explained and multiplexing issues are described.

Author(s)	:	M.M.G.N. van den Bergh
Reviewers	:	ir. F.C.I. van den Eijnden, ir. P.H.A. Venemans
Department	:	Communication Architectures and Open Systems (CAS)
Project	:	ATM Pilot
Project manager	:	Ir. J.C. van der Wal
Project number	:	71650
Commissioned by	:	N NO PCS
Date	:	July 1995

Classification: Only for use within KPN and Eindhoven University of Technology

Person responsible at KPN Research: Head of department CAS

Key words:
ATM, Quality of Service, Network Performance, TCP, UDP, IP, ATM pilot, performance experiments, multiplexing

Mailing list: Alg. dr KPN Research, dr R&D KPN Research, mt CAS, dr.ir. H.J.M. Bastiaansen, dr. J.L. van den Berg, prof. ir. J. de Stigter, ir. A.J.F. van Halderen, ir. F.H. Leerkes, dr. ir. R.J.F. de Vries, ir. F.W. de Vries, ir. P.B.J. Oude Vrielink, drs. L.J. Teunissen, ir. M.J.G. Dirksen, ir. A.C. Doelman, ir. G.P. Buitenhuis, ir. M.V. Boersma, ing. J.P. Humpig, ir. J. de Bie, dr. ir. M. de Graaf, ir. A.A.L. Reijnierse, ir. H. Sewberath Misser, drs. M.J.E. Raffali, leden ATM Pilot project, ir. F.C.I. van den Eijnden (5x), auteur (10x)

Management Summary

The development of the Asynchronous Transfer Mode (ATM) technology gives rise to a number of new communication services, such as high-speed data exchange between distantly located local area networks (LANs), high-resolution picture transfer, high quality interactive videotex, and in particular videophone, video conference, and distributive TV, all in addition to the conventional voice services.

Many telecommunication services make use of an underlying protocol named TCP/IP. Secondly, ATM technology is being applied to local area networking, where it offers greatly increased bandwidth and supports broadband services. However, ATM's success as a LAN technology depends on its ability to provide LAN-like services compatible with existing protocols, like TCP/IP, and applications. Therefore the interworking between TCP/IP and ATM forms a critical issue in the development of commercially attractive ATM services.

In order to test ATM services in practise and to achieve experience in running an operational ATM network, PTT Telecom NL started a national ATM pilot in July 1994. Furthermore, this pilot provides the opportunity to obtain the experience needed to verify and complete the international B-ISDN/ATM standards provided by the ATM Forum and the International Telecommunication Union (ITU).

In this report, the gained experience with the available ATM equipment and ATM traffic control functions is described. The interworking between the TCP/IP protocol and ATM is extensively studied and the influence of several parameters on the performance of TCP/IP over ATM is examined, performing a number of ATM network experiments. Limits for each parameter are extracted from the performed experiments. It can be concluded that the performance of TCP/IP over ATM complies with the expectations on basis of theoretical considerations, as long as the measured parameter-limits are not exceeded. Moreover, recommendations are given regarding a possible adaption of the TCP/IP protocol, the available ATM equipment at KPN Research and items for further study. This report can be used in order to provide reliable ATM services in the near future.

Summary

Asynchronous Transfer Mode (ATM) is widely acknowledged as the base technology for the next generation of global telecommunications. One of the first opportunities for network operators to exploit the ATM network commercially, is to offer TCP/IP connections over ATM in the near future. This report describes the examination of the performance of TCP/IP over an operational ATM network.

The basic principles of ATM, as far as needed in this report, are described and the distinction between the Quality of Service (QoS) and Network Performance (NP) is explained. ATM is able to achieve better utilization of the network resources through multiplexing of traffic streams. However, gains due to (statistical) multiplexing come at risk of potential cell loss or extra cell delay and might affect the NP and QoS.

Before TCP/IP connections over ATM can be offered, the interworking between the TCP/IP protocol and ATM has to be extensively studied and experience with the available ATM equipment and ATM traffic control functions has to be gained. Therefore, in this report the following topics are considered during ATM network experiments:

- The segmentation of User Data into UDP datagrams or TCP segments and the translation into ATM cells via AAL5.
- The influence of peak rate limitation at the data sources on the throughput of UDP/IP and TCP/IP over ATM.
- The impact of the datagram size (MTU) on throughput of UDP/IP and TCP/IP over ATM.
- The consequences of cell loss for the throughput of TCP/IP over ATM.
- The influence of extra traffic parameters on the throughput of TCP/IP over ATM.
- The intention to perform multiplex experiments.

Generally, it can be concluded that, for sufficiently low peak rate settings at the data source, a large MTU size, and a low Cell Loss Ratio, the throughput of UDP/IP and TCP/IP over ATM complies with the expectations on basis of theoretical considerations. The impact of smaller MTU sizes and cell loss on the throughput is caused by extra processing delay, consisting of the interdatagram gap, intercell gap and an extra delay component due to cell loss. The results can be used in order to provide reliable TCP/IP connections over ATM in the near future.

Preface

This report forms the result of my final Master's thesis, performed as part of my study of Electrical Engineering at the Eindhoven University of Technology. Between October 1994 and August 1995 I worked at the Dr. Neher Laboratory of KPN Research in Leidschendam. For me this was an opportunity to experience the way of working in a big company and doing applied research on both theoretical and practical aspects of Asynchronous Transfer Mode.

Within my work I was offered the challenge to develop and display a maximum of creativity in order to fully utilize the limited available equipment. In the beginning, this sometimes resulted in a disappointment, but the satisfaction is enormous now I succeeded.

At this point, I would like to take the opportunity to thank prof.ir. F. van den Dool for taking the responsibility for my graduation project, his interest in my progress and his useful comments. Also I thank my coach K. van der Wal for his neverending benevolence to discuss my ideas and problems and whom I have experienced as an inexhaustible source of knowledge and experience. Thank to all the members of the ATM Pilot project and other colleagues for the helpful discussions and remarks, both about my graduation project and future life. Next to them, the other students doing their graduation project or practical traineeship added greatly to the pleasant time I had at KPN Research. Finally I thank my parents for encouraging me to study and for supporting me in every possible way.

July 1995
Marc van den Bergh

Table of contents

List of figures

List of tables

Abbreviations

AAL	:	ATM Adaptation Layer
ACR	:	Average Cell Rate
ARP	:	Address Resolution Protocol
ATC	:	Adaptive Traffic Controller
ATM	:	Asynchronous Transfer Mode
BER	:	Bit Error Rate
B-ISDN	:	Broadband Integrated Services Digital Network
BL	:	Bucket Limit
BT	:	Burst Tolerance
CAC	:	Connection Admission Control
CBR	:	Constant Bit Rate
CCITT	:	International Consultative Committee for Telephone and Telegraph
CDV	:	Cell Delay Variation
CER	:	Cell Error Rate
CLP	:	Cell Loss Priority
CLR	:	Cell Loss Rate
CMR	:	Cell Misinsertion Rate
CPP	:	Cell Protocol Processor
CS	:	Convergence Sublayer
CSMA/CD	:	Carrier Sense Multiple Access with Collision Detection
CTD	:	Cell Transfer Delay
FIFO	:	First In First Out
FTP	:	File Transfer Protocol
HEC	:	Header Error Correction
IETF	:	Internet Engineering Task Force
IP	:	Internet Protocol
ITU-T	:	International Telecommunication Union - Telecommunication Standardisation Sector

LAN	: Local Area Network
LIFO	: Last In First Out
LLC	: Logical Link Control
MAC	: Medium Access Control
MCTD	: Mean Cell Transfer Delay
MSS	: Maximum Segment Size
MTU	: Maximum Transmission Unit
NP	: Network Performance
PCR	: Peak Cell Rate
PDH	: Plesiochronous Digital Hierarchy
PDU	: Protocol Data Unit
PFB	: Police Function Board
QoS	: Quality of Service
RTT	: Round Trip Time
SAR	: Segmentation and Reassembly sublayer
SCR	: Sustainable Cell Rate
SECBR	: Severely Errored Cell Block Ratio
STM	: Synchronous Transfer Mode
TAXI	: Transparent Asynchronous Transmitter/Receiver Interface
TCP	: Transmission Control Protocol
UDP	: User Data Protocol
UPC	: Usage Parameter Control
VBR	: Variable Bit Rate
VC	: Virtual Channel
VCC	: Virtual Channel Connection
VCI	: Virtual Channel Identifier
VP	: Virtual Path
VPC	: Virtual Path Connection
VPI	: Virtual Path Identifier
WTSC	: World Telecommunication Standardisation Conference

1 Introduction

1.1 Problem definition and purpose

Asynchronous Transfer Mode (ATM) is widely acknowledged as the base technology for the next generation of global telecommunication. ATM provides the flexibility to integrate the transport of a wide mix of services including voice, video and data, with a broad range of bandwidth requirements. This high degree of resource sharing results in many possible traffic mixes.

The users (customers) demand a certain quality of a telecommunication service and are not concerned with any of the aspects of the networks internal design. The network provider, on the other hand, tries to meet the user requirements, but also tries to exploit the telecommunication network as efficient as possible. The Quality of Service pertains to the user oriented performance-concerns of an end-to-end service, while Network Performance is concerned with parameters needed for network planning, operations and maintenance.

ATM is able to gain better utilization of the network resources through multiplexing of variable bit rate traffic streams. Gains due to statistical multiplexing, however, come at risk of potential cell loss. Also the buffering that must occur in each multiplexing and switching point in the network may introduce delay. Which of these consequences are relevant, depends on the type of service that is being considered. Therefore, in order to achieve both higher network utilization and also fulfil the user requirements, the network providers must develop a better understanding of ATM traffic characteristics and the traffic control functions in practise.

PTT Telecom NL has started a national ATM pilot in parallel to the European ATM pilot. The main goals of this pilot are to achieve experience in running an operational ATM network and to test ATM services in practise.

Since TCP/IP* applications are widely used, network operators are interested in the opportunity to offer TCP/IP connections over ATM in the near future. Before these services can be offered, experience with the available ATM equipment and ATM traffic control functions has to be gained and the influence of several settings and parameters on the performance of TCP/IP over ATM should be examined. Therefore, in this report a number of ATM network experiments is performed and analyzed. Besides the performance of a single TCP/IP connection over ATM, also the usefulness of multiplexing of TCP/IP traffic is examined, taking the influences on the Network Performance and Quality of Service into account.

1.2 Report outline

Chapter 2 starts with an introduction to Asynchronous Transfer Mode (ATM) in order to provide the reader with enough relevant information for understanding the rest of the report. In chapter 3 the terms Quality of Service and Network Performance are explained, QoS and NP parameters are defined and their relationship is discussed. Two principles of multiplexing, peak rate allocation and statistical multiplexing, are described in chapter 4. Issues like traffic parameters, complexity of traffic control functions and the required buffer size are taken into account. Chapter 5 is devoted to the interworking between TCP/IP and ATM. LAN services like TCP/IP can be emulated over ATM, but TCP/IP can also be mapped directly on AAL5. In chapter 6 a layered performance model for TCP/IP applications over ATM is described, in which the influence of QoS parameters on the NP parameters and vice versa is examined. Also the efficiency and throughput of TCP/IP over ATM is considered. Chapter 7 is devoted to the ATM network experiments. Either NP parameters, traffic control parameters or parameters at IP or TCP level are varied, while the performance is measured. Multiplexing experiments are described in chapter 8. Due to a small output buffer in the present GDC switch, the usefulness of statistical multiplexing could not be examined. However, solutions for this problem are given and future multiplex experiments are recommended. Chapter 9 completes this study on the performance of TCP/IP over an operational ATM network with conclusions and recommendations.

* *When TCP/IP is mentioned, both TCP/IP and UDP/IP are intended, except for the network experiments described in chapter 7 and 8, where TCP/IP and UDP/IP are clearly distinguished.*

2 Asynchronous Transfer Mode

2.1 Introduction

In the broadband ISDN an extremely wide variety of applications with different traffic characteristics and performance requirements is expected.
Asynchronous Transfer Mode (ATM) [Pry91], [Vries94] as the principle network switching, transmission and multiplexing technique for B-ISDN, aims at the following favourable properties:

- Flexible and efficient user access (varying information rates are allowed).
- True integration of all information services for transmission, multiplexing and switching in a uniform manner (all services can be handled by the same network resources).
- Rapid support of new services.

The ATM bearer service is able to integrate any connection-oriented service such as circuit-switched type services with constant bit rates, as well as services with highly variable bit rates, as computer, video, and packetized voice communications.

The asynchronous transfer mode is a fast packet switching technique based on virtual connections using small fixed-size packets called cells. A cell consists of 53 octets, 5 of which are reserved for the cell header and 48 for user information. Each header contains, among others, a virtual channel and a virtual path identifier (VCI, VPI), payload and priority indicators, and one octet for a one-bit header error forward correction and for the self-delineation of cell boundaries. Cell sequence integrity is preserved per virtual channel.

As ATM is connection-oriented, connections are established for the duration of a call, i.e. a call can be divided into three phases: call set-up phase, information transfer phase and call termination phase. At call set-up, routes through the network are determined, VPI(s) and VCI(s) are allocated, and network resources are reserved.

During the information transfer phase, user cells follow the routes that have been established and at call termination all claimed VPI(s), VCI(s) and resources are freed. No network capacity is consumed by a connection unless cells are actually being transported. There is only limited processing on the cell's way through the network, e.g. there is no error and flow control on a link-by-link basis. Hence, high information transfer speeds can be achieved. Signalling and user information are carried on separate virtual channels.

This promising integration technique, however, raises a number of new problems due to the higher degree of resource sharing compared to the conventional synchronous transfer mode STM. Some of these questions are highly dependent on the user or traffic source characteristic and therefore must be considered on the basis of stochastic traffic flows.

The strength of ATM is that the network gives the control to fill time slots with cells to the user. At the same time this is the weakness of ATM. When the network has no more control over the filling of time slots, the network has no more control over the amount of traffic entering the network. The total traffic offered by the users may exceed the network capacity. As a result the links could be overloaded and the buffers may overflow resulting in high cell losses and exorbitant delays. This situation is known as congestion. Proper traffic control and resource management capabilities are needed to keep congestion within acceptable limits, while trying to make a maximum and efficient use of network resources.
However, since ATM is able to support a wide variety of bit rates and varying bit rates, it is hard to predict what kind of traffic B-ISDN terminals will produce. In addition the traffic mix, e.g. the superposition of several traffic sources with different characteristics is even harder to predict.

2.2 ATM Adaptation layer

The service provided by the ATM layer, i.e. the transfer of information fields from a source to one (or more) destination(s), is a generic transfer service which can not directly be used by current higher layer protocols. Since, the modification of well defined standard higher layer protocols is troublesome or undesired, an intermediate layer between the ATM layer and the higher layers is needed: the ATM Adaptation Layer (AAL). The main purpose of this layer is to isolate the higher layers from the specific characteristics of the ATM layer by mapping the higher layer Protocol Data Units (PDUs) into the information fields of the ATM cells and vice versa. The basic function, which can be recognized at this point, is the segmentation and reassembly of the higher layers PDUs.

This function is very important and, hence, it justifies a separate logical sublayer: the SAR (Segmentation And Reassembly sublayer). The SAR is positioned directly above the ATM layer and is the lower of the two logical sublayers of the AAL.
The functionality of the SAR may not be sufficient for all higher layer protocols. Extra convergence functions are required which can be used to provide a more sophisticated service to the AAL service user. These sublayers are located in the second logical sublayer of the AAL, the Convergence Sublayer (CS). The CS is positioned directly below the higher layers and is inherently service dependent.

2.3 Service classes

In principle, any imaginable application can be supported using ATM, irrespective of the transfer requirements. A way to prevent an uncontrolled growth of service offering and to minimise the number of protocols used in B-ISDN is the definition and standardisation of service classes in the AAL.
The three key parameters, that have been used to derive the service classes, are:

- Time relation; a timing relation between source and destination(s) can be required or not required.
- Bit rate; the bitrate can be constant (CBR) or variable (VBR).
- Connection mode; two connection modes can be identified: connection-oriented and connectionless.

Other parameters, such as assurance of the communication, are treated as QoS parameters and do not lead to further (sub)classification of the AAL. Since not all combinations of the key parameters are foreseen, four service classes are distinguished, class A to D. Table 1 shows the definition of these classes, i.e. the relation between the classes and the key parameters.

■ ***Table 1*** *Service classification for the AAL.*

	Class A	*Class B*	*Class C*	*Class D*
Time relation	*Required*		*Not required*	
Bit rate	*Constant*	*Variable*		
Connection mode	*Connection-oriented*			*Connectionless*

Examples of telecommunication services in each of the classes are:

- Class A: circuit emulation, constant bit rate video.
- Class B: variable bit rate video and audio.
- Class C: connection oriented data transfer.
- Class D: connectionless data/packet transfer.

For every (higher layer) class of services there exist a certain combination of functions inside the AAL and these combinations are mapped onto AAL types. CCITT* Recommendation I.363 describes a number of AAL types (protocols) which consist of combinations of SAR and CS functions and can support higher layer services belonging to one of the above defined classes. Up to now four AAL types have been elaborated in ITU-T. AAL Type 1 is directed to class A services, AAL Type 2 to class B services and AAL Type 3/4 (formerly separate as AAL Type 3 and AAL Type 4) to class C and D services. AAL Type 5 has been included recently for high speed data services.

More detailed information about ATM and the AAL can be found in [Pry91] and [Vries94].

* *At the World Telecommunication Standardisation Conference (WTSC) of March 1993 in Helsinki, the CCITT has been renamed to International Telecommunication Union (ITU).*

3 Quality of Service and Network Performance

3.1 Introduction

At least two parties are involved in a telecommunication service. The users (customers) demand a certain quality of a telecommunication service and are not concerned with any of the aspects of the networks internal design. Secondly, the network provider tries to meet the user requirements, but also wants to exploit the telecommunication network as efficient as possible.

3.2 Terminology

It is generally accepted that Quality of Service (QoS) is the user's view of a service as opposed to the provider's view. Definition of QoS, however, remains a difficult task because of the varying factors, such as:

- Many different types of users.
- Many different types of applications.
- Subjective dependence on the users view of the service.

Despite this fact, Quality of Service (QoS) is defined in ITU-T Recommendation I.350 as follows: "Collective effect of service performances which determine the degree of satisfaction of a user of the service".

Network performance (NP) is measured in terms of parameters which are meaningful to the network provider and are used for the purposes of system design, configuration, operation and maintenance. NP is defined independently of terminal performance and user actions.

The QoS has a direct relationship to the NP as shown in figure 1.

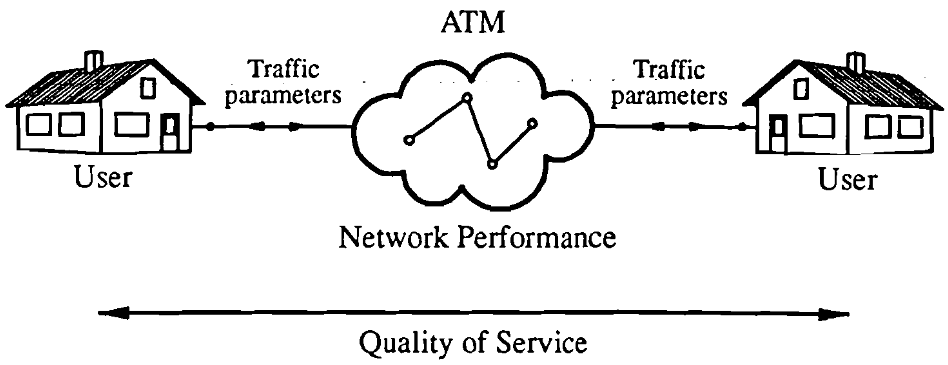


Figure 1 General reference configuration for QoS and NP.

According to ITU-T Recommendation I.350, "the user oriented QoS parameter values provide a valuable framework for network design, but they are not directly usable in specifying performance requirements for particular connections. Similarly, the NP parameter primarily determine the QoS, but they do not necessarily describe that quality in a way that is meaningful to users".

The principle difference is that QoS pertains to user oriented performance concerns of an end-to-end service, while NP is concerned with parameters that are of concern to network planning, provisioning and operations activities. However, also QoS has to be considered from the beginning of network design to meet the user's requirements for efficient communications. Some of the characteristics which distinguish QoS and NP are depicted in table 2.

■ *Table 2 Distinction between QoS and NP.*

Quality of Service	*Network Performance*
User oriented	*Provider oriented*
Service attribute	*Connection element attribute*
Focus on user-observable effects	*Focus on planning, development (design), operation and maintenance*
Between (at) service access points	*End-to-end or network connection elements capabilities*

A typical user is not concerned with any of the aspects of the networks internal design. He is interested only for the satisfaction of his demands.

From the user's point of view, QoS may be best expressed by parameters [I.350] which:

- Focus on user-perceivable effects, rather than their causes within the network.
- Should not depend on the internal design of the network.
- Have to take into account all aspects of the service from the user's point of view, which can be objectively measured at the service access point.
- May be granted to a user at the service access point by the service provider.
- Are described in network independent terms and create a common language understandable by both the user and the service provider.

A network provider is concerned with the efficiency and effectiveness of the network, in providing services to customers. Therefore, from the network provider's point of view, network performance is best expressed by parameters which provide information for:

- System development.
- Network planning, both nationally and internationally.
- Operation and maintenance.

In the following, ATM traffic at several time levels is considered and taking this into account QoS and NP parameters are defined.

3.3 Traffic control aspects

In an ATM network traffic control operates at several time levels; the cell level, the activity level and the connection level, see figure 2. Each level has its own time scale dealing with some network control aspects.

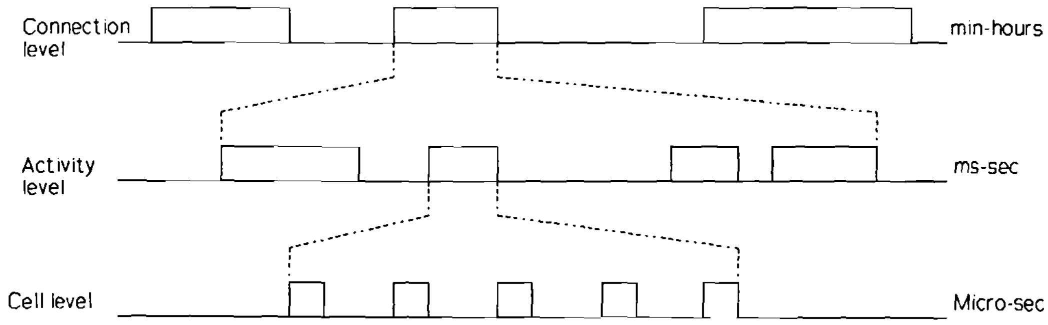


***Figure 2** Traffic flows at three time scales.*

3.3.1 The cell level

The cell level operates at a time scale in the order of micro seconds. At this level the individual cells of the virtual connections can be seen. Congestion at the cell level occurs when several cells contend for the same link. Only one cell can be placed on this link, while the other cells have to wait in a buffer. The buffer has to be dimensioned in such way that the probability of buffer overflow is sufficiently small. However, a large buffer can result in a large queuing delay and more variation in delay (jitter), depending on the traffic in the network.

3.3.2 The activity level

The time scale of the activity level lies in the order of milli seconds to seconds. At this level the activity of the traffic source can be seen. Some sources transmit a burst at regular intervals, constant bit rate (CBR) sources. Other sources do not transmit bursts regularly during the whole call, they consist of active and idle states. These sources are said to be bursty at the activity level and transmit at a variable bit rate (VBR). ATM can gain bandwidth efficiency by statistical multiplexing of bursty traffic sources. Since a bursty source does not require continuous allocation of the bandwidth at the peak rate, a large number of bursty sources can share the link capacity, while the link capacity is less than the sum of the peak rates. This is further explained in paragraph 4.4.

Congestion at the activity level occurs when a number of traffic sources transmit a burst simultaneously so that the total sum of the required bandwidths exceed the link capacity. The buffers in the network are not intended to deal with this activity level congestion, because of delay sensitive applications. In stead the call acceptance control must take care that the probability that too many sources transmit at the same time, is kept below a certain threshold.

3.3.3 The connection level

At the connection level the time scale ranges from minutes to several hours. At this level the start and duration of connections can be observed. Congestion at the connection level is experienced as a blocking of calls when the network is almost fully loaded. The decision whether new connection still can be admitted depends on the status of the network, the traffic characteristics of the new connection and the desired Quality of Service. If a path through the network can be found from source to destination with enough resources available to meet these criteria, the call is accepted.

Congestion at the connection level has to be solved by network planning, i.e. proper dimensioning of buffer space, link capacity, topology design, etc.

3.4 Quality of Service parameters

Each time scale deals with some Quality of Service concerning call set-up and transfer phase. ITU-T distinguishes at least the following Quality of Service parameters [I.371], [Dirk91]: cell loss, end-to-end delay, variation in delay and call blocking. These aspects are essential in order to allow:

- The customer to know what to expect from the service.
- The network operator to allocate the necessary resources to the service (connection acceptance control).
- The network operator to check that the customer is not exceeding the agreed limits (source policing), potentially degrading the services offered to the other customers of the network.

Each QoS aspect is now discussed and related to the appropriate time scale.

- ***Cell loss***
  Cell loss occurs whenever buffers in the switches and multiplexers are overloaded. At the cell level this may happen due to the asynchronous arrivals of cells if in a time interval more cells arrive than can be served and placed on the outgoing link. At the activity level cell loss may occur when too many users transmit at the same time resulting in an offered load exceeding the link and buffer capacity. Summarizing, cell loss at the cell level and activity level is caused by buffer overflow.

- ***End-to-end delay***
  End-to-end delay is the summation of packetization delay, propagation delay and queuing delay. According to [Dirk91], the main cause for delay is not the queuing delay in switching nodes, but the propagation delay through the links and (for low bit rates) the packetization delay to fill a cell with data. The maximum queuing delay occurs when a completely filled buffer is encountered in all switching stages. This is unlikely to happen. Except for queuing delay, all delays are fixed once a connection is established.

- ***Cell delay variation***
  CBR services expect cells to arrive at regular intervals at the destination. Therefore cell delay variation is undesirable and has to be kept as low as possible. Queuing delay is the cause for cell delay variation. Cells experience different queuing delays due to asynchronous cell arrivals of other cell streams delay variation takes place at the cell level.

- ***Call blocking***
  Call blocking typically occurs at the connection level. A connection request is rejected when not enough resources are available to guarantee the agreed Quality of Service for the established connections any more when this new connection would be allowed. Resources include buffers, bandwidth, processing power, etc., but lack of bandwidth is the most important cause for call blocking.

Different application types require a different QoS in order to function satisfying. In the following the relationship with the QoS parameters involved is examined for a number of applications. The QoS parameters considered, regarding the mentioned applications, are loss tolerance and delay tolerance (including delay variation).
First a list of interesting applications is presented:

- File transfer (FTP).
- Speech (interactive).
- Video (interactive).
- LAN interconnect.
- Data distribution.
- Video/audio distribution.
- E-mail.
- Real-time transaction processing.
- High speed data communication.

This list certainly is not comprehensive. Figure 3 shows the relationship between the mentioned applications and the QoS parameters.

		Delay Tolerance		
		Low	*Middle*	*High*
Loss Tolerance	*Low*	* *Video* * *High speed data communication* * *Real-time transaction processing*	* *E-mail* * *File transfer* * *LAN interconnect*	* *Video/audio distribution*
	High	* *Speech*		* *Data distribution*

Figure 3 *Relationship between applications and QoS parameters.*

3.5 Network Performance parameters

This section defines a set of ATM cell transfer performance parameters, according to ITU-T Recommendation I.356.

- ***Cell error ratio***
  Cell error ratio (CER) is the ratio of total errored cells to successfully transferred cells plus errored cells in a population of interest. Successfully transferred cells and errored cells contained in cell blocks counted as severely errored cell blocks should be excluded from the population used in calculating cell error ratio.

- ***Cell loss ratio***
  Cell los ratio (CLR) is the ratio of total lost cells to transmitted cells in a population of interest. Lost cells and transmitted cells in cell blocks counted as severely errored cell blocks should be excluded from the population used in calculating cell loss ratio.

- ***Cell misinsertion rate***
  Cell misinsertion ratio (CMR) is the total number of misinserted cells observed during a specified time interval divided by the time interval duration (or equivalently, the number of misinserted cells per connection second). Misinserted cells and time intervals associated with cell blocks counted as severely errored cell blocks should be excluded from the population used in calculating cell misinsertion rate.

- ***Severely errored cell block ratio***
  Severely errored cell block ratio (SECBR) is the ratio of total severely errored cell blocks to total cell blocks in a population of interest.

- ***Cell transfer delay***
  Cell transfer delay (CTD) is the time, t_2-t_1, between the transmitting of a cell at time t_1 and the receiving of that cell at time t_2, where $t_2 \geq t_1$.

- ***Mean cell transfer delay***
  Mean cell transfer delay is the arithmetic average of a specified number of cell transfer delays.

- ***Cell delay variation***
  If the cells belonging to a single cell stream are monitored, a cell-to-cell variation in network transit delay can be observed. Depending on the measurement method 1-point Cell Delay Variation (CDV) and 2-point cell delay variation can be defined. 1-Point CDV is defined on basis of a reference clock and the monitoring of a sequence of successive cells at a single measurement point, whereas for 2-point CDV corresponding arrivals at two measurement points are involved. Further explanation can be found in [I.356].

3.6 QoS classes

A user of an ATM connection (a VCC or a VPC) is provided with one of a number of QoS classes supported by the network. It should be noted that a VPC may carry VC links of various classes, but the QoS of that VPC must meet the most demanding QoS of the VC links. The QoS class associated with a given ATM connection is allocated to the network at the time of connection establishment and will not change for the duration of that ATM connection. QoS class can have specified performance parameters (Specified QoS class) or no specified performance parameters (Unspecified QoS class). Specified QoS Class provides a quality of services to an ATM virtual connection (VCC or VPC) in terms of a subset of the ATM performance parameters defined in the previous section. In a Specified QoS class, at most two cell loss ratio parameters may be specified. If a Specified QoS class does contain two cell loss ratio parameters, the one parameters is for all CLP=0 cells and the other parameter is for all CLP=1 cells of the ATM connection. ITU has decided to use one bit in the ATM cell header for explicit Cell Loss Priority (CLP) indication. When this bit is set, a cell has low priority with regard to loss, i.e. such a cell is subject to discarding depending on network overload and congestion conditions.

The following Specified QoS classes are currently defined, using the service classes that are defined before:

- Specified QoS class 1: support a QoS that will meet Service Class A performance requirements.
- Specified QoS class 2: support a QoS that will meet Service Class B performance requirements.
- Specified QoS class 3: support a QoS that will meet Service Class C performance requirements.
- Specified QoS class 4: support a QoS that will meet Service Class D performance requirements.

In the unspecified QoS class, no objective is specified for the performance parameters. However, the network provider may determine a set of internal objectives for the performance parameters and these objectives need not to be constant during the duration of a call. An example of the Unspecified QoS class is the support of "best effort" service.

3.7 Sources of QoS degradation

The following list of items can influence the QoS in a network:

- ***Propagation delay***
  This is the delay caused by the physical medium which transports the bits between end-points and ATM switches, dependent upon the distance only.

- ***Media Error Statistics***
  This is the random and/or bursty bit errors that are introduced on the physical medium.

- ***Switch architecture***
  The overall architecture of the switch can have significant impacts on the performance. Some aspects to consider are the switch matrix design (from blocking to non-blocking), buffering strategy (input-, output or central buffering) and the switch characteristics under load (FIFO, LIFO, priorities).

- ***Buffer capacity***
  This is the actual capacity of the buffer in units of cells.

- ***Number of tandem nodes***
  This is the number of ATM switching nodes that a particular VPC or VCC traverses.

- ***Traffic load***
  This is the load offered by a set of ATM VPC/VCCs on the same route as the VPC/VCC under consideration.

- ***Resource allocation***
  This is the capacity allocated to a VPC/VCC.

- ***Failures***
  These are events that impact availability, such as port failures, switch failures or link failures. Switch overs between failing equipment or circuits may introduce cell loss.

3.8 Impact of degradation sources on performance parameters

In this section the impact of each of the sources of QoS degradation from the previous section on each of the Network Performance parameters is analyzed [ATMF 94]. Changes in the Network Performance parameters might influence the Quality of Service parameters. This is further explained in paragraph 6.5.

- ***Cell error ratio and Severely errored cell block ratio***
  The Cell error ratio (CER) and the Severely errored cell block ratio (SECBR) are expected to be primarily influenced by the error characteristics of the physical media and in case of SECBR also by buffer overflows.

- ***Cell loss ratio***
  The Cell los ratio (CLR) is expected to be influenced by errors in the cell header, buffer overflows, and the non-ideal User Parameter Control (UPC) actions. Cells may also be lost due to failures, protection switching and path reconfiguration. The number of nodes will impact the CLR due to the possibility of overflow in any buffer between the source and destination.

- ***Cell misinsertion rate***
  Cell misinsertion ratio (CMR) is expected to be primarily influenced by undetected/miscorrected errors in the cell header.

- ***Cell transfer delay***
  Cell transfer delay (CTD) is affected by propagation delay, queuing, routing and switching delays.

- ***Mean cell transfer delay***
  Mean cell transfer delay will likely be dominated by propagation delay, queuing, routing and switching delays.

- ***Cell delay variation***
  The Cell delay variation will be primarily influenced by the buffering and the traffic load.

The following table summarizes how various sources of degradation can impact the Performance parameters.

■ ***Table 3*** *Influence of degradation sources on NP parameters.*

	CER	*CLR*	*CMR*	*MCTD*	*CDV*
Propagation delay				*X*	
Media Error Statistics	*X*	*X*	*X*		
Switch architecture		*X*		*X*	*X*
Buffer capacity		*X*		*X*	*X*
Number of tandem nodes	*X*	*X*	*X*	*X*	*X*
Traffic load		*X*	*X*	*X*	*X*
Failures		*X*			
Resource allocation		*X*		*X*	*X*

4 Multiplexing

4.1 Introduction

Asynchronous Transfer Mode (ATM) provides a structure in which cells from different connections are multiplexed or switched using a common fabric, independent of the connections's bit rates or burstiness. ATM connections can be allocated resources based on either peak rate allocation or statistical multiplexing [Pry91], [Vries94].
In order to achieve satisfying multiplexing results, it is necessary for the user to describe the expected traffic as close as possible, using some traffic parameters. Multiplexing can improve the efficient utilization of network resources, but should not affect the agreed QoS.

4.2 Traffic parameters

In the traffic contract between the user and the network operator some traffic parameters and QoS parameters are given. Using the traffic parameters for the Connection Admission Control and policing function, the network provider is able to fulfil the demanded QoS. The QoS parameters define the requirements for the network and are therefore translated into Network Performance parameters.
Traffic parameters describe traffic characteristics of an ATM connection. A traffic parameter is a specification of a particular traffic aspect. It may be quantitative or qualitative. These parameters may for example describe Peak Cell Rate, Average Cell Rate, Sustainable Cell Rate, Burstiness, Burst Tolerance and source type.

- ***Peak Cell Rate***
  According to Recommendation I.371, the Peak Cell Rate (PCR) traffic parameter specifies an upper bound on the traffic that can be submitted on an ATM connection. Enforcement on this bound by the UPC allows the network operator to allocate sufficient resources to ensure that the Network Performance objectives can be achieved. The PCR of the ATM connection is the inverse of the minimum inter-arrival time T and T is called the Peak Emission Interval of the ATM connection.

- ***Average Cell Rate***
  The Average Cell Rate (ACR) is the number of cells transmitted divided by the duration of the connection.

- ***Sustainable Cell Rate***
  The Sustainable Cell Rate (SCR) is an upper bound on the conforming average rate of an ATM connection. Enforcement on this bound by the UPC could allow the network operator to allocate sufficient resources, but less than those based on the PCR, and still ensure that the Network Performance objectives can be achieved.

- ***Burstiness***
  No general accepted definition of burstiness has been described yet. The burstiness at cell time scale here is defined as the PCR divided by the ACR. A continuous cell stream arrival pattern (CBR) has by definition a burstiness equal to 1.

- ***Burst Tolerance***
  Given the PCR, the SCR and a time interval t, the Burst Tolerance is the time period that a source is allowed to transmit at the peak rate, so that the SCR in the time interval t is not exceeded. This is depicted in the following figure.

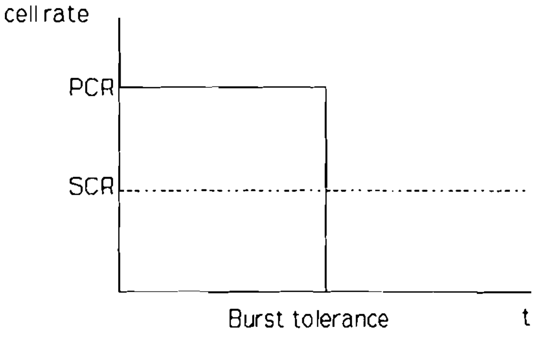


***Figure 4** Burst Tolerance.*

- ***Source type***
  A number of traffic classes will be defined. The user has to declare what kind of source will be used.

The PCR must be specified for every connection. The SCR and Burst Tolerance traffic parameters are optional traffic parameters a user may choose to declare jointly, if the user can upper bound the realized average cell rate of the ATM connection to a value below the PCR. The SCR and Burst tolerance traffic parameters enable the user to describe the future cell flow in greater detail than just the PCR. In that case the network provider may be able to utilize the network resources more efficiently.

4.3 ATM peak rate allocation

The most simple strategy of multiplexing is to reserve bandwidth corresponding to the agreed peak bit rate of the connection, called peak rate allocation. Peak bit rate is defined here as the maximum bit rate at which the user agreed to transmit.
With peak rate allocation, the sum of the peak bandwidths for the constituent connections is less than or equal to the peak bandwidth capability of the channel into which they are multiplexed. An ATM network that uses only peak rate allocation can be designed with a limited set of traffic management controls. Congestion at activity level will not be an issue at peak rate allocation, since the admission control assures that the link capacity is not exceeded. However, limited buffering is required at multiplexing and switching points to accommodate the effects of cell level congestion, i.e. in case two cells arrive at the multiplexer at the same time, one cell has to be buffered. This causes cell delay variation (CDV). In this mode, decisions on the admission of new connections can be based purely on the quantity of unallocated bandwidth in the network, so the important traffic parameters are the peak cell rate and the CDV tolerance.
Furthermore ATM peak rate allocation provides better network utilization over traditional circuit-switched connections for the following reasons:

- Less bandwidth granularity. the bandwidth dedicated to a particular service can be more precisely matched to the service's needs.
- More flexibility and integration of all information services.
- Single-stage multiplexing and switching. There is no need to dedicate bandwidth to the different hierarchical levels like at PDH transmission. The same bandwidth pool is available to all services.

On the negative side for ATM is the additional bandwidth required for the ATM header and ATM adaptation layer functions.
A large number of sources do not transmit at peak rate continuously, but have an average bit rate lower than the peak bit rate, for example LAN-LAN datatransmission. The unused capacity, however, can not be used by other services since peak rate capacity has been reserved for that particular source, see figure 5.

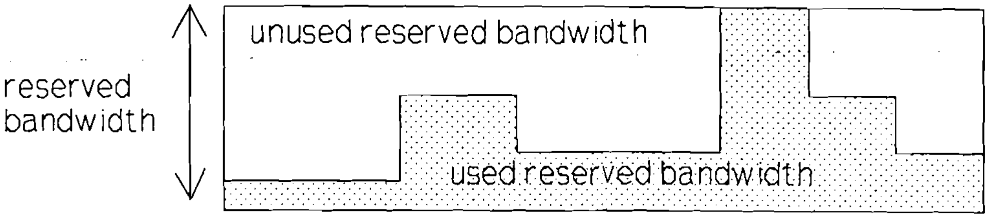


***Figure 5** Peak rate allocation.*

4.4 ATM statistical multiplexing

As mentioned before, peak rate allocation results in a poor utilization of bandwidth for bursty sources. Statistical multiplexing occurs when the capacity of an output channel is less than the sum of the peak connection bandwidths, but is normally larger than their average total bandwidth requirement. The statistical gain is the factor by which the sum of the peak bandwidths exceeds the output channel's capacity, i.e. the sum of the peak rates divided by the link capacity. Statistical multiplexing, therefore, relies on the input channels being bursty due to variable information transfer rates. Hence, the statistical gain directly depends on the bandwidth utilization and the traffic characteristics of the input channels.

ATM allows for statistical multiplexing at higher levels than the cell level. Achieving any statistical gain results in a non-zero probability of cell-level overload (congestion at cell level) or congestion at activity or connection level.

According to ITU-T Recommendation I.371, in B-ISDN, congestion is defined as a state of network elements (e.g. switches, concentrators, cross-connects and transmission links) in which the network is not able to meet the negotiated network performance objectives for the already established connections and/or for the new connection requests. In general, congestion can be caused by

- Unpredictable statistical fluctuations of traffic flows.
- Fault conditions within the network.

Congestion is to be distinguished from the state where buffer overflow is causing cell losses, and thus affects the NP, but still meets the negotiated Quality of Service.
The left side of figure 6 shows two traffic streams (A and B) that will be statistical multiplexed.

The traffic flow after multiplexing a number of connections is given on the right side in figure 6.

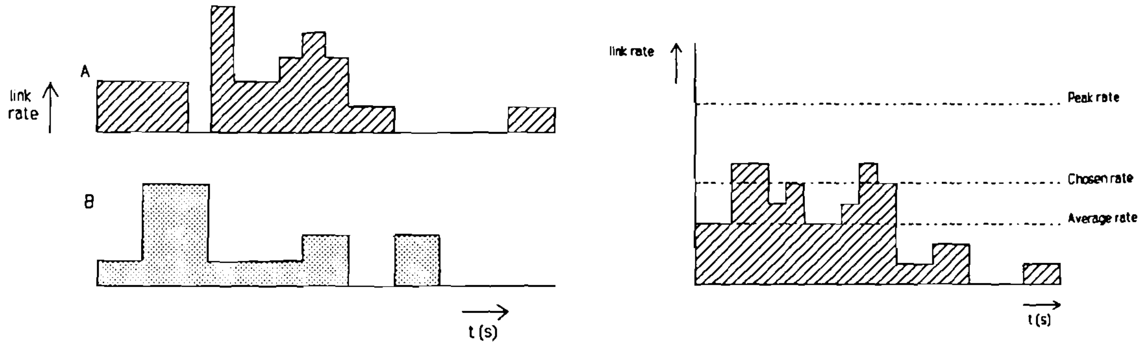


***Figure 6** Required link capacity reduction by statistical multiplexing.*

In case of peak rate allocation, enough link capacity is reserved for every situation. However, it is also possible to allocate less capacity than the sum of the peak rates (statistical multiplexing). Now the probability that many sources transmit at peak rate at the same time and the allocated capacity is exceeded, is assumed to be very small. Buffers are used to temporarily store traffic that exceeds the allocated link rate. In case small buffers are implemented the cell loss probability due to buffer overflow is still large, but the extra delay and delay variation due to buffering are small. For larger buffers, the cell loss probability is smaller, but extra delay and delay variation can occur.

When infinite buffers are implemented, the allocated capacity can be decreased to the average rate. In this situation, however, the cell delay and the CDV can become very high. The user or operator will define a maximum delay, what results in limited buffering and in a certain amount of capacity to allocate. The 'chosen rate' in figure 6 indicates this amount of allocated capacity. In [Dirk91] this chosen rate is called equivalent bandwidth, i.e. the minimal amount of bandwidth necessary for a connection to still meet the QoS objectives.

Yet, it is very difficult to determine how many sources can be statistically multiplexed without unacceptable loss of quality, since this depends on other users and the QoS parameters. Not for all sources statistical multiplexing will be successful. According to [Dirk91], sources with the following characteristics can be successfully multiplexed resulting in high statistical gain:

- The peak rate should be about 10% or less of the link capacity.
  The probability that all sources transmit at the peak rate should be very small, so the number of sources to be multiplexed should be large.

- The peak to mean bit rate ratio should be large.
  In that case statistical multiplexing is more efficient.

- The length of the burst must be relatively short with respect to the length of silence periods. This is important in case the duration of congestion should be limited. The length of a burst will influence the distribution of the cell loss probability.

A clear distinction should be made between the multiplexing gain and the number of sources that can be multiplexed into a link. To show the difference between these two, consider the following example:

■ **Example 1**

Suppose there is a link with a capacity of 100 Mb/s. In case there are sources with a peak rate of 20 Mb/s and an average rate of 5 Mb/s, only 5 sources can be multiplexed using peak rate allocation and about 75% of the capacity remains unused. Using statistical multiplexing and infinite buffers, in total 20 sources can be multiplexed ($S.M._{max}$) and the total available capacity is used. However, in this situation cell delay can become very large. Figure 7 shows a relationship between the queue length (buffer size) and the bandwidth that can be allocated.
Suppose that the user decides that the delay should be limited and because of that a maximum queue length is defined, which results in a certain amount of allocated bandwidth (equivalent bandwidth); in case of figure 7 this is 10 Mb/s. Now statistical multiplexing of 10 sources still offers the QoS that will satisfy the user ($S.M._{sat}$), while the statistical multiplexing gain is 2.

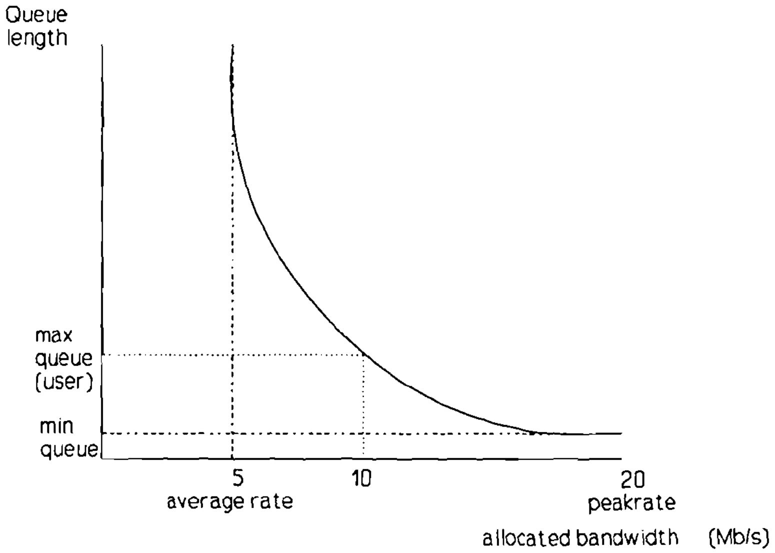


Figure 7 Relation between queue length and allocated bandwidth.

In case the peak rate at the sources decreases and the average rate remains the same, the difference between peak rate allocation and statistical multiplexing decreases. Also the multiplexing gain decreases, while on the other hand more sources can be multiplexed in a link. This is depicted in the table 4.

■ ***Table 4*** *Difference between multiplexing gain and number of sources in a link.*

Peak Rate (Mb/s)	*Av. Rate (Mb/s)*	*P.R. Alloc. (# sources)*	*S.M.$_{max}$ (# sources)*	*S.M.$_{sat}$ (# sources)*	*Gain*
20	*5*	*5*	*20*	*10*	*2.0*
10	*5*	*10*	*20*	*15*	*1.5*
7	*5*	*14*	*20*	*17*	*1.3*
5	*5*	*20*	*20*	*20*	*1.0*

From table 4 it can be concluded that for Constant Bit Rate no statistical gain is achieved, but a maximal number of sources can be multiplexed.

■

The previous example indicates that the network provider should not only focus on the statistical gain. Sources with lower peak rates by applying for example traffic shaping will also improve efficient utilization of the available network capacity.

Traffic control function, like Connection Admission Control, policing and congestion control become more complex in case of statistical multiplexing and more traffic parameters than the peak cell rate and the average cell rate are needed.

4.5 Buffer size versus cell loss probability

The importance of the cell loss probability strongly depends on the kind of application. In case of speech communication a relative large amount of cells can be lost [Wal94], but the cell loss probability has to be low for data transfer applications. Figure 3 in section 3.4 shows the cell loss dependence of several applications.
Applying statistical multiplexing results in a non-zero probability of cell loss. This is no problem as long as this cell loss probability is very low or the QoS requirements of the application are still met. In case the cell loss requirements can not be met any more, the traffic load should be decreased or buffers should be implemented, which might result in extra delay.

Figure 8 shows that the cell loss probability first decreases rapidly when the buffer size is increased, but after a certain buffer length this probability barely decreases any more [Dirk91]. It appears that the curve of cell loss probability as function of the buffer size can be distinguished into two regions; the cell level region and the activity level region.

At cell level the buffers are needed to deal with simultaneous cell arrivals.
When too many sources are active at the same time the required bandwidth may exceed the link capacity. Buffers at the activity level could be used to deal with these bursts.

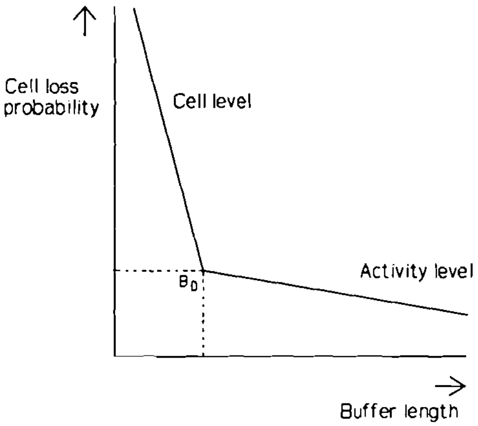


Figure 8 *Cell loss probability versus buffer size.*

In the cell level region the cell loss probability decreases rapidly when the buffer size is slightly increased. However, in the activity level region, the cell loss probability decay is much smaller when the buffer length is increased. The position of breakpoint B_0 depends on the burst length and the number of sources. Using simulation and analyses results, it appears that the buffer size at the breakpoint is generally smaller than 100 cells ([Dirk91]). When considering buffer dimensioning of an ATM switch several arguments have to be taken into account.

- The buffers in the ATM switching nodes are intended to deal with the random cell arrivals due to statistical multiplexing, not for bursts.

- In the activity level region an enormous amount of extra buffers is needed to significantly lower the cell loss probability, see figure 8.

- The size of the buffers should be as small as possible to keep the cell delay variation low. Cell delay variation is a problem for applications that expect a constant bit rate at the destination. In case of extremely large buffers the probability of large cell delay increases. For applications that are delay sensitive, this might be a problem.

These reasons argue for a buffer size in the order of magnitude of B_0. In the situation that the network operator decides to offer delay insensitive services that do not require constant bit rate, it is possible to implement larger buffers.

Presently also switches are produced containing two (or more) buffers. CBR and high priority VBR traffic with a low delay tolerance is placed into the small buffer (in the order of magnitude of B_0), while medium or low priority traffic with a higher delay tolerance is placed into a much larger buffer.

4.6 Traffic control functions

A number of functions and mechanisms help in managing the traffic offered to the network and providing the QoS required by the customers. The traffic control functions that are considered, are Connection Admission Control (sometimes also referred to as Connection Acceptance Control) and policing (in ITU-T terminology Usage/Network Parameter Control). An additional function that may be used is traffic shaping. Most of these functions and mechanisms have been standardised in Recommendation I.371. In this section the increased complexity of these mechanisms and the extra traffic parameters that are needed in case of statistical multiplexing are described.

4.6.1 Connection Admission Control

The connection admission control (CAC) decides whether there are sufficient resources available to accept a new connection, while the agreed QoS of the already existing connections must be preserved. At connection set-up the network and the user first have to agree on a traffic contract in which the bandwidth needed for the connection is declared. The user must provide the network with the values of a number of traffic parameters and the desired QoS parameters. Based on these values and the load state of the network, the CAC decides whether the requested connection can be accepted or has to be rejected.

The CAC must be able to deal with many possible traffic mixes, i.e. situations where CBR and VBR connections with varying characteristics share network resources. Furthermore, the CAC should try to optimise network efficiency, e.g. by adopting statistical multiplexing gain in case of VBR connections. On the other hand CAC should be kept simple in order to enable real-time response to connection set-up requests. In ITU-T there is a consensus that CAC procedures should be network operator specific [I.371] (network efficiency and QoS offered to the user is the responsibility of the network operator).

In case of peak rate allocation, only the Peak Cell Rate (PCR) is part of the traffic contract. When statistical multiplexing is applied, traffic parameters like Sustainable Cell Rate (SCR) and Burst Tolerance are also needed.

On the basis of these parameters CAC can perform bandwidth reservation on a VBR basis and, thus, VBR traffic contracts become possible. However, it has to be noticed that, if one goes beyond the Peak rate allocation, CAC becomes a rather complex matter. Further discussion of CAC is outside the scope of this report.

4.6.2 Policing

Policing is the monitoring of traffic at the cell or activity level in order to determine whether user cell streams behave as has been agreed upon at call set-up, and the invocation of specific actions in case of violations or user misbehaviour. Its main purpose is to protect network resources from malicious as well as unintentional misbehaviour which can affect the QoS of other already established connections.
In the situation of peak rate allocation, the policing function only needs to make sure that the agreed PCR is not exceeded, e.g. by using a leaky bucket.
The leaky bucket consists of a counter representing the bucket level. The bucket level is incremented each time a cell is passed into the network, i.e. the cell rate of the customer. Meanwhile the counter is periodically decremented, using the PCR.
When the source transmits cells at a higher rate, the counter value increases up to a certain maximum, the bucket limit BL. When the bucket limit has been reached, successive cells are discarded until the bucket contents has sufficiently leaked away.

When statistical multiplexing is applied, both the peak cell rate and the sustainable cell rate need to be policed, so two leaky buckets are needed.
The policing of the PCR remains the same as during peak rate allocation.
The second leaky bucket level is incremented by the traffic cell rate and decremented by the SCR. The SCR can be exceeded for a certain time (Burst Tolerance), but then the bucket limit has been reached. It is clear that applying statistical multiplexing, requires more traffic parameters and a more complex policing mechanism.

4.6.3 Traffic shaping

Traffic shaping is a function by which the traffic characteristics of a stream of cells belonging to a particular connection are changed in order to achieve a desired modification of those characteristics. Traffic shaping may be optionally performed at the entrance of the network, for example, to reduce the peak cell rate.
In stead of statistical multiplexing it might also be possible to shape VBR traffic to CBR traffic using a buffer and apply peak rate allocation.

4.7 Usefulness of statistical multiplexing

In this section, the advantages, disadvantages and open issues of statistical multiplexing will be discussed.
For a network provider, the main advantage of statistical multiplexing is the possibility of high resource utilization, i.e. less capacity will remain unused and the performance of the network* will improve. The user, however, should not experience any decrease of the Quality of Service.

The most important disadvantage of statistical multiplexing is the fact that it becomes more complex for the network provider to guarantee the required QoS to the user in all situations, i.e. more traffic parameters are needed. As shown in the previous section, traffic control functions like the Connection Admission Control and the policing become more complex and require more parameters.

Another open issue is whether statistical multiplexing is needed in the first place, since in future maybe bandwidth is not scarce at all any more because of optical fibers or efficient coding techniques.

For many applications the multiplexing gain might be small and for constant bit rate traffic there is no gain at all. Only in case of bursty traffic statistical multiplexing will be interesting. In order to guarantee the agreed QoS, buffers should be implemented in the switches and multiplexers. These buffers should be large enough to guarantee the agreed cell loss ratio, but should be small enough to guarantee the agreed cell delay.

Finally, it can be concluded that statistical multiplexing is an interesting technique in theory, but it requires complex control functions and network parameters that it will be questionable whether statistical multiplexing can be implemented in practise.

Since statistical multiplexing is expected to be mainly interesting for data transmission, in the next chapter data transfer applications using TCP/IP or UDP/IP over ATM are deepened out.

* *The performance of a network has different meaning than the Network Performance. In fact the term NP might be confusing, since the performance of only one connection is considered (see paragraph 3.2). The term 'Connection Performance' might be more clear. With the term 'performance of a network' the efficiency and effectiveness of a number of links is indicated.*

5 TCP/IP over ATM

5.1 Introduction

For network operators it is very interesting to be able to offer TCP/IP connections over ATM in the near future, since TCP/IP applications are widely used. First more information is obtained about the TCP/IP protocol suite and the interconnection with ATM. LAN services can be emulated over ATM (LAN emulation), but network layer protocols, like IP, can also be mapped directly on AAL5. Later in chapter 6, the translation of QoS parameters at TCP/IP application level to ATM NP parameters is considered.

5.2 TCP/IP protocol suite

The TCP/IP architecture [Feit93] was designed in the 1970s by the United States Defence Advanced Research Projects Agency (DARPA), in order to connect equipment from different vendors. The TCP/IP architecture glues clusters of networks together, creating a larger network called an internet. To a user, an internet simply appears to be a single network, composed of all the hosts connected to any of the constituent networks. The protocols were required to be independent of host hardware or operating system and to be robust, surviving high network error rates. In order to give communication software a structure that is rather simple and easy to modify, a layered structure is attractive.

Figure 9 shows how parts of the TCP/IP protocol suite fit together, also compared to the OSI architecture. The underlying layers in the figure are the Datalink layer and the Physical layer. In TCP/IP the distinction between the network layer and the transport layer is critical; the network layer (IP) provides a hop-by-hop service, while the transport layers (TCP and UDP) provide an end-to-end service.

Two styles of application-to-application interactions are required:

- Connection-oriented communication is appropriate when the applications need a sustained interchange of streams of data, including error correction (TCP is used).
- Applications engaged in connection-less communication exchange standalone messages, without error correction (UDP is used).

The TCP/IP protocol suite includes a set of standard applications including file transfer (FTP), remote login (Telnet), and electronic mail (e-mail). It also has become routine to offer a remote printing service.

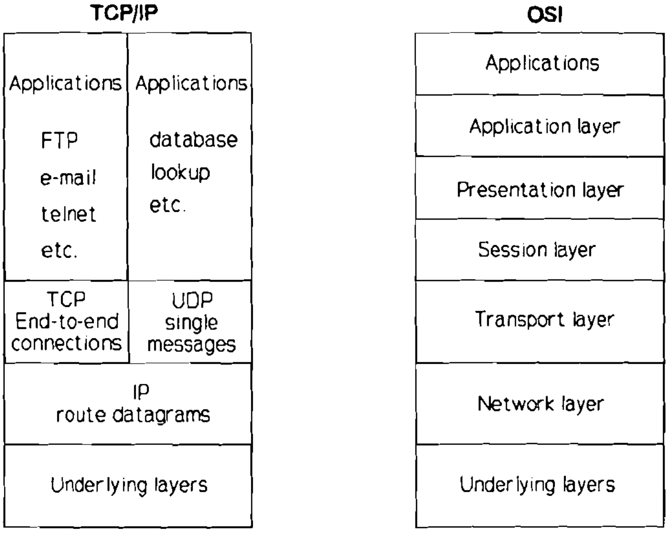


Figure 9 *TCP/IP and OSI architecture.*

The OSI model uses the term Layer N Protocol Data Unit (PDU) for the information unit that the Nth layer protocol deals with. A PDU consists of a header and (optionally) some enclosed data. Peer-to-peer communication cooperate by exchanging PDUs. TCP/IP uses different terms for the information units at a certain layer. Rather than saying Layer 4 PDU or Layer 3 PDU, the TCP, UDP and IP terms are *segment* or *UDP Datagram* at layer 4, and *datagram* at layer 3. At Application Layer, data is referred to as *User Data.* A lower layer protocol data unit is called a *frame.* A frame header contains fields that identify the source and destination physical devices.

5.3 Internet Protocol

The Internet Protocol (IP) routes data between hosts. IP is an interworking protocol developed by the Department of Defense in the US in the early 1970s. IP is an example of a connectionless service.

It permits the exchange of traffic between the two host computers without any prior call setup. (However, these two computers usually share a common connection-oriented transport protocol). It is possible that data units could be lost between the two end users' stations, since IP is an unreliable, best effort, data unit-type protocol, and it has no reliability or error correction mechanisms. It provides no recovery of errors at the underlying subnetworks, it has no flow-control mechanisms, the user data (data units) may be lost, duplicated or arrive out of order.

It is not the job of IP to deal with these problems. Most of the problems are passed to higher-level transport layers, like TCP or UDP. As a result, IP is reasonably simple to install and, because IP routing is done on a hop-by-hop basis, it is quite robust in a way that a new route is selected when the old route is (partly) unavailable. The IP protocol implements two basic functions: addressing and fragmentation. An IP datagram is made up of an IP header and a unit of data to be delivered. The internet modules use the addresses carried in the internet header to transmit internet data units towards their destinations. The selection of a path for transmission is called routing.

The internet modules use fields in the internet header to fragment and reassemble internet data units when necessary for transmission through "small packet" networks. The header length is measured in 32-bit words. Normally, the size for an IP header is 5 words (20 octets), but optionally the header length can be 15 words (60 octets). The length of a datagram is restricted by the Maximum Transmission Unit (MTU).

The performance of an internet depends on the quantity of available resources in its hosts and routers and on how efficiently the resources are used. These resources are:

- Transmission bandwidth.
- Buffer size.
- CPU processing.

Protocol design involves tradeoffs between gains and losses in efficiency. Some positive performance features of IP are bandwidth sharing, dynamic rerouting, little control overhead, good buffer usage, simple processing and fixed address size. However, IP has also some negative features, like limited flow control, route at each hop and reassembly overhead.

5.4 User Data Protocol

The User Data Protocol (UDP) is a simple, datagram-oriented transport layer protocol. Applications invoke UDP in order to send isolated messages to each other, i.e. connectionless communication. The overhead of sending and receiving the many messages required to set up and take down a connection is avoided by simply sending a query and a response. UDP packages data into units called UDP datagrams and passes them to IP for routing to a destination. Each output operation by a process produces exactly one UDP datagram, which causes one IP datagram to be sent. This is different from a stream-oriented protocol such as TCP where the amount of data written by an application may have little relationship to what actually gets sent in a single IP datagram.

Since IP is unreliable, there is no guarantee of delivery. If an application sends a query in a UDP datagram and a response does not come back within a reasonable amount of time, it is up to the application to retransmit the query or not. The application needs to worry about the size of the resulting IP datagram. If it exceeds the network's MTU, the IP datagram is fragmented. When an IP datagram is fragmented, each fragment becomes its own packet, with its own IP header, and is routed independently of any other packet. This makes it possible for the fragments of a datagram to arrive at the final destination out of order, but there is enough information in the IP header to allow the receiver to reassemble the fragments correctly. However, if one fragment is lost, the entire datagram must be retransmitted. IP has no timeout and retransmissions, that is the responsibility of the higher layers. TCP performs timeout, retransmission and error correction, but UDP does not. (Some UDP applications perform timeout and retransmission themselves). The IP fragmentation will be further examined in paragraph 7.4.2 and 7.5.2.

UDP is a connectionless service that often is used for simple database lookups, or for constructing monitoring, debugging, management and software testing functions. The UDP header consists of 8 bytes.

5.5 Transmission Control Protocol

Even though the Transmission Control Protocol (TCP) and UDP use the same network layer (IP), TCP provides a totally different service to the application layer than UDP does. TCP provides a connection-oriented, reliable, byte-stream service. TCP contains mechanisms to guarantee that data is error-free, complete, and in sequence.

5.5.1 TCP services

The user data is broken into what TCP considers the best sized chunks to send. This is totally different from UDP, where each write by the application generates a UDP datagram of that size. The unit of information passed by TCP to IP is called a *segment.* When TCP sends a segment it maintains a timer, waiting for the other end to acknowledge reception of the segment. If an acknowledge is not received in time, the segment is retransmitted. This strategy is called positive acknowledgement with retransmission. When TCP receives data from the other end of the connection, it sends an acknowledgement. TCP maintains a checksum on its header and data. This is an end-to-end checksum whose purpose is to detect any modification of the data in transit.

If a segment arrives with an invalid checksum, TCP discards it and does not acknowledge receiving it. (It expects the sender to time out and retransmit, see figure 10). Since TCP segments are transmitted as IP datagrams, and since IP datagrams can arrive out of order, TCP segments can arrive out of order. A receiving TCP resequences the data if necessary, passing the received data in the correct order to the application. Besides, since IP datagrams can get duplicated, a receiving TCP must discard duplicate data. TCP also provides flow control. Each end of a TCP connection has a finite amount of buffer space. A receiving TCP only allows the other end to send as much data as the receiver has buffers for.

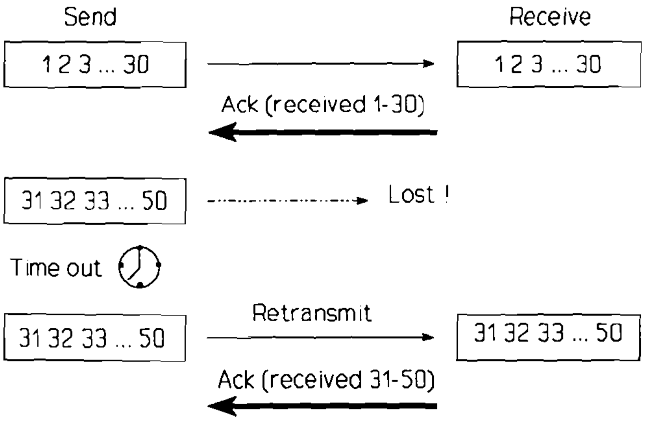


Figure 10 *TCP timeout and retransmission.*

Services such as file transfer or electronic mail are built on top of TCP. The normal size of the TCP header is 20 octets, unless options are present. TCP headers only carry an option during connection set up, and so it is reasonable to assume that the header will consist of 20 octets. Normally the IP header will also consist of 20 octets.

Large segments give better performance during bulk data transfers, because a smaller percentage of bandwidth and memory resources are used for headers. During connection setup, each party can declare the Maximum Segment Size (MSS) that it is willing to receive, i.e. the maximum amount of data that can be carried in a segment. The size of the TCP header is not included in the MSS value. In choosing segment sizes, maximum datagram limits (MTU size) also must be considered. The maximum MSS value is the MTU, minus the size of the TCP and IP headers.

5.5.2 Flow control

The TCP data receiver is in charge of its incoming flow of data, the transmitter has to adapt to the receiver's limits. During connection setup, each partner assigns receive buffer space to the connection. This number is usually an integer multiple of the MSS. Incoming data flows into the receive buffer and stays there until it is absorbed by the application. Buffer space is used up as data arrives. When the receiving application removes data, space is cleared for more incoming data. This sliding window protocol is fundamental to the efficient transfer of bulk data.

Receive window
The receive window consists of any space in the receive buffer that is not occupied by data. Data will remain in a receive buffer until the targeted application accepts it. The receive window extends from the last acknowledged byte to the end of the buffer. When the application accepts the data in the receive buffer, space for new incoming data becomes available. This is visualized by sliding the window to the right. Every ACK sent by the receiver contains an update on the current state of its receive window. The flow of data from the sender is regulated according to these window updates.

Send window
The data transmitter maintains a send buffer that tracks how much data has been sent and acknowledged, and the size of the partner's receive window. The send buffer extends from the first unacknowledged octet to the right edge of the current receive window. The send window covers the unused part of the buffer.
The initial sequence number and initial receive window size are announced during connection setup. A copy of the bytes sent, must be kept in the send buffer until the bytes have been acknowledged, since they may have to be retransmitted.

Slow start and congestion window
Slow start is the way to initiate data flow across a connection. At some point packets can be dropped and congestion avoidance is a way to deal with lost packets. The assumption of the algorithm is that packet loss caused by damage is very small (much less than 1%), therefore the loss of a packet signals congestion somewhere in the network between the source and destination.

There are two indicators of packet loss: a timeout occurring and the receipt of duplicate ACKs. A duplicate ACK is an immediate acknowledgement when an out-of-order segment is received.

Congestion avoidance and slow start are independent algorithms with different objectives. But when congestion occurs, the transmission rate of packets into the network is slowed down and then slow start is invoked to get things going again. In practise they are implemented together. Congestion avoidance and slow start require that two variables are maintained for each connection: a congestion window, *cwnd*, and a slow start threshold size, *ssthresh*. The combined algorithm operates as follows: Initialization for a given connection sets *cwnd* to one segment and *ssthresh* to 65535 bytes. The TCP output routine never sends more than the minimum of *cwnd* and the receiver's advertised window.

Congestion avoidance is flow control imposed by the sender, while the advertised window is flow control imposed by the receiver. The former is based on the sender's assessment of perceived network congestion; the latter is related to the amount of available buffer space at the receiver for this connection.

When congestion occurs (indicated by a timeout or the reception of duplicate ACKs), one-half of the current window size (the minimum of *cwnd* and the receiver's advertised window, but at least two segments) is saved in *ssthresh*. Additionally, if congestion is indicated by a timeout, *cwnd* is set to one segment (i.e. slow start).
When new data is acknowledged by the other end, *cwnd* is increased, but the way it is increased depends on whether slow start or congestion avoidance is performed.
If *cwnd* is less or equal to *ssthresh*, slow start is performed; otherwise congestion avoidance is performed. Slow start continues until halfway the size of the window at the moment congestion occurred (this value was saved in *ssthresh*), and then congestion avoidance takes over.

Congestion avoidance dictates that *cwnd* is incremented by 1/*cwnd* each time an ACK is received. This is an additive increase, compared to slow start's exponential increase. This way *cwnd* is increased at most one segment each round trip time (RTT), while slow start will increment *cwnd* by the number of ACKs received in a round trip time.

Fast retransmit and fast recovery algorithms

The purpose of the duplicate ACK is to let the other end know that a segment was received out of order, and to tell it what sequence number is expected. Since it is unknown whether a duplicate ACK is caused by a lost segment or just a reordering of segments, the retransmission algorithm waits for a small number of duplicate ACKs to be received. If three or more duplicate ACKs are received in a row, it is a strong indication that a segment has been lost.

Then retransmission of the missing segment is performed, without waiting for the retransmission timer to expire. This is the fast retransmit algorithm. Next, congestion avoidance, but not slow start is performed. This is the fast recovery algorithm.
The reason for not performing slow start in this case is that the receipt of the duplicate ACKs indicates that there is still data flowing between the network ends, since the receiver can only generate the duplicate ACK when another segment is received.

5.6 Interworking between IP and ATM

Much of the interest in ATM stems from the promise of its vastly increased bandwidth and greater flexibility and manageability. However, its success as a LAN technology depends on its ability to provide LAN-like services compatible with existing protocols and applications, like TCP/IP. Therefore both the network and service providers are interested in providing TCP/IP applications over an ATM network.

5.6.1 LAN service requirements

Presently, LANs offer connectionless, best effort, service for the transfer of variable size data packets [New94]. LANs offer point-to-point, multicast, and broadcast transfer. Many current protocols rely on the broadcast capability. Users are not required to establish a connection before submitting data for transmission, nor are they required to define traffic characteristics of their data in advance of transmission. Users simply submit traffic to the LAN whenever they wish, as fast as possible, and the LAN dynamically shares the available bandwidth between all active users.
Most LAN equipment conforms to the IEEE 802 family of protocols (see figure 11*). In this architecture, the data link layer is split into the Logical Link Control (LLC) and Medium Access Control (MAC) sublayers.

The LLC sublayer offers a common interface to the network layer, while each different MAC protocol is specified to a particular LAN, e.g. Carrier Sense Multiple Access with Collision Detection (CSMA/CD), Token Ring, Token Bus, etc. Resolving the IP address (or the address of another network layer protocol) to MAC address is done by an Address Resolution Protocol (ARP).

LANs are frequently interconnected with bridges and routers to form larger networks. Bridges operate at the MAC sublayer, and are popular because they require very little manual configuration and are transparent to the user. Bridges interconnect multiple LAN segments yet give the appearance to the user of a single LAN.

* *It is not intended to map the ATM broadband reference model directly on the OSI model.*

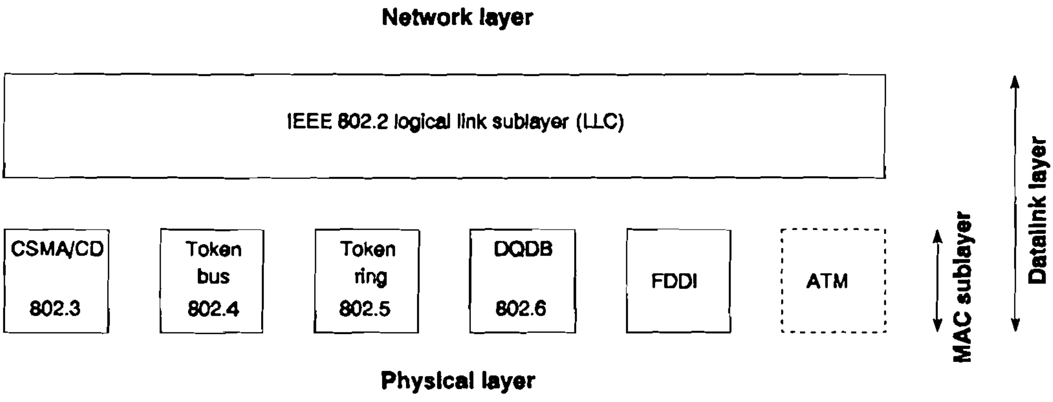


Figure 11 The IEEE 802 family of LAN protocols.

Routers operate at the network layer but support only a finite set of network layer protocols. They offer greater control, better management facilities, and may be used to construct larger networks than bridges.

In contrast to IEEE 802 LAN's, ATM networks are inherently connection oriented. With respect to IP and other network layer protocols, ATM can be configured either as a separate data link protocol or as a MAC protocol below the LLC. The former approach results in IP and other network protocols to be implemented directly over ATM. The latter approach is the key idea behind LAN emulation. The next figure shows LAN emulation compared to IP over ATM. This figure also shows the co-called "native ATM applications", that are designed to fit on top of the AAL. This of course is the final goal, but at the moment manufacturers of ATM products also have to take the legacy LAN protocols into account [Voo94].

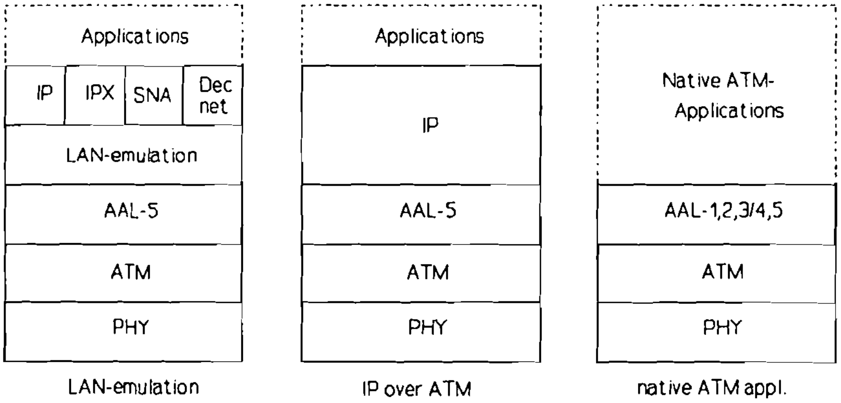


Figure 12 LAN emulation, IP over ATM and native ATM applications.

It is important to realise that the three stacks above are different. LAN emulation is more interesting for PCs and PC-manufacturers like IBM, while manufacturers of powerful workstation, such as Fore Systems and SUN chose for a strategy with IP over ATM. Lan emulation and IP over ATM will probably coexist in the future.

5.6.2 LAN emulation

LAN emulation ([Voo94], [Chao94], [New94], [Jeff94]) simply means that the point-to-point ATM switch should give the appearance of a virtual shared medium. From the point of view of the protocol stack, the ATM layer should behave like yet another IEEE 802 MAC protocol below the Logical Link Control (LLC). The key attribute of all shared medium interconnects is that all communication is broadcast, implying that all stations in a LAN receive all the packets, and they filter out the packets that they want to receive. Although ATM is connection oriented, the broadcast feature can be emulated in an ATM network using dedicated servers.

The LAN Emulation Sub-working Group of the ATM Forum has identified a number of different servers which include the LAN Emulation (LE) server and one or more multicast servers. The LE server provides functionality for registering and resolving MAC addresses and/or route descriptors to ATM addresses. The multicast servers are required to provide the connectionless data delivery characteristics of a shared network to hosts that are directly connected to the ATM network, referred to as the LE Clients. One key function that the multicast server should support is the ability to multicast a MAC frame to a set of LE Clients. Given this functionality, LAN emulation can be simply supported by requiring all LE Clients to send all the MAC frames and ARP queries and replies to the multicast server, which can broadcast them to the LE Clients. In this approach, the LE Clients are responsible for selectively filtering out the packets they want to receive.

In order to reduce the amount of traffic in the network, the ATM switch and the multicast server can be architected in such a manner that only the ARP queries are processed by the multicast server. All other data transfers use the point-to-point connections established across the ATM switch.

The functions required for this purpose are the ability to resolve MAC addresses to ATM addresses, referred to as LE-ARP and the ability to perform connection management. This is further explained in an example, using an ATM LAN, shown in figure 13.

■ **Example 2**

If Host A wants to send an IP packet to host B, only the IEEE 802 LAN segments get involved. If A wants to send an IP packet to D and does not know D's MAC address, it first broadcasts an ARP request, which is received by bridge B2. The bridge sends the ARP query to all the valid outgoing ports, which in this case is the port connected to the ATM switch and more specifically that of the multicast server. The server broadcasts the ARP query to all the LE Clients. When D gets the ARP request, it responds by trying to establish a point-to-point connection to A. In the LE-ARP server, A's MAC address is associated with bridge B2's ATM address. So D establishes a point-to-point connection to bridge B2 and sends the ARP reply. Bridge B2 forwards the ARP reply to A and binds D's MAC address to the appropriate port. When A sends an IP packet in a MAC frame with D's MAC address, bridge B2 receives the frame and forwards it to the appropriate port.

■

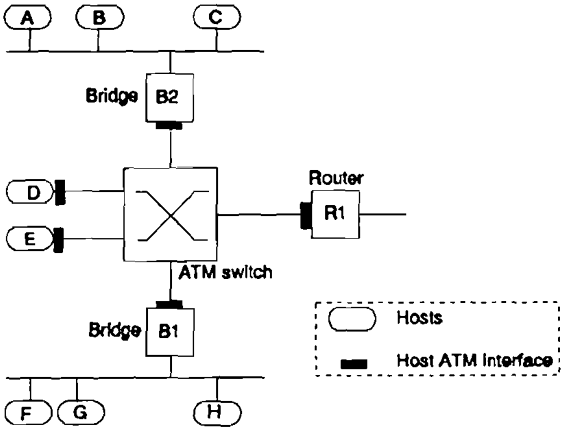


Figure 13 A local ATM network.

Since LAN emulation defines the ATM layer as a MAC protocol below the LLC, supporting IP over an emulated LAN is the same as supporting IP over any IEEE 802 LAN. The key issue is to define a LAN emulation architecture that is both scalable and flexible in the sense that it can support the majority of the well established network layer protocols efficiently. The LAN emulation architecture should be able to handle not only unicast data transfer, but also broadcast and multicast data transfers. This architecture is currently being standardized by the LAN Emulation Sub-working Group [LASG94].

5.6.3 IP over ATM

Instead of emulating LAN services over ATM, it is also possible to map network layer protocols, like IP, directly on AAL5, using an approach similar to LAN emulation at the MAC sublayer. The major difference is that the address resolution mechanism must translate directly from the network layer address, e.g. IP address to the ATM address.

The straight forward approach is to maintain a server, referred to as the IP-ATM-ARP server. The IP-ATM-ARP server needs to maintain tables that can translate an IP address to an ATM address. The interaction between the hosts and the IP-ATM-ARP server can be implemented using a simple query/response protocol. Mapping network layer protocols directly on AAL5 means legacy LAN protocols are no longer necessary. The disadvantage of direct IP over ATM is that only one network layer protocol is supported (in this case IP), whereas LAN emulation offers support to various network layer protocols (e.g. IP, IPX, SNA, OSI, Decnet). The advantage of direct IP over ATM is that it is simpler, and that is characterised by less overhead. The implementation of IP over ATM is being standardised by the Internet Engineering Task Force (IETF) in RFC 1577: Classical IP and ARP over ATM.

In the rest of the report IP over ATM is considered, since in the ATM pilot, IP is mapped directly on AAL5. Because LAN Emulation is not included in the ATM pilot , the examination of this approach in practise is outside the scope of this report

6 Layered Performance Model

6.1 Introduction

Generally, in order to guarantee the agreed QoS, it is necessary to translate user requirements into NP parameters for the ATM network. QoS requirements can impose strict constraints on some ATM NP parameters. On the other hand, the influence of changes of NP parameters to the QoS parameters at application level will be considered. Therefore a QoS framework for TCP/IP over ATM is defined and all operations and features at each layer that might influence the QoS are discussed. Finally the efficiency and throughput of TCP/IP over ATM is examined.

6.2 QoS Framework

The performance requirements not only depend on the diverse QoS requirements from the user, but also on the layer processing and workload in host systems. So, in order to offer the user a certain QoS, it is necessary to introduce a set of QoS parameters whose properties indicate the nature and the requirements in a layered model. These QoS parameters are defined for each layer such that each layer can guarantee the demanded QoS to its next-higher layer and demand a (possibly different) QoS from its next-lower layer [Jung93]. On the other hand, errors or actions taken at a certain layer may influence the QoS parameters of higher layers. The proposed QoS framework consists of the QoS parameters, the NP parameters and the layered model. The QoS and NP parameters at ATM level have been introduced in section 3.4 and 3.5.

Figure 14 shows a layered model for direct TCP/IP over ATM, including each layer's data unit name. The data is segmentized by the TCP protocol and transferred to datagrams at the IP layer. The maximum size of a datagram is specified by the MTU. The AAL5 cuts the IP datagram into a number of ATM cells. The number of IP datagrams needed to transmit a file of a certain number of bytes, mainly depends on the set MTU size. TCP adds a twenty bytes header, while UDP only adds eight bytes. Then also IP adds a header of twenty bytes.

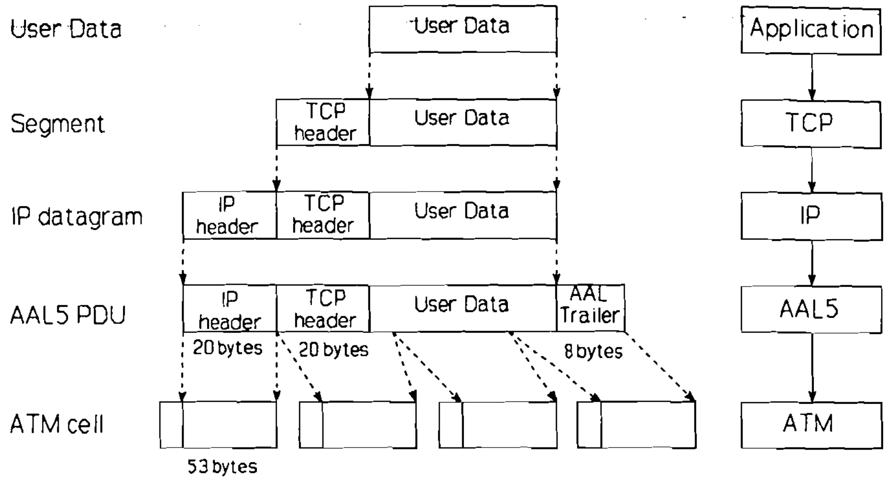


Figure 14 Protocol structure of TCP/IP over ATM.

The rest of the MTU can be filled with data information. In case the file to be sent is larger than the MTU, the file is segmented into several datagrams. If an internet datagram is fragmented, its data portion should be broken on eight octet boundaries, according to [RFC791]. The translation of User Data in IP datagrams and ATM cells is further deepened out in paragraph 7.4.2 and 7.5.2.

6.3 QoS parameters at data transfer application level

The QoS parameters at application level indicate the way the user experiences an application as explained in section 3.4. In case of TCP/IP and UDP/IP applications, the following QoS parameters are considered:

- ***Transfer delay***
  The transfer delay is the value of elapsed time between the start of transfer and successful transfer of a specified amount of User Data.

- ***Throughput***
  The throughput is defined by the amount of successfully transferred User Data divided by the transfer delay. The throughput is measured in bits per second.

In case of TCP/IP applications information loss will not occur, because of retransmissions at a lower level. However, these retransmissions cause extra delay and decrease the throughput. Retransmitted data is not included as User Data, and therefore not considered in the throughput.

In case of UDP/IP applications one other QoS parameter should be considered:

- ***Information Loss***
  Information Loss is defined as the amount of lost User Data measured in bytes.

6.4 Influence of bit errors on NP at ATM level

In this paragraph the influence of random bit errors at the physical layer on the ATM NP parameters is examined, i.e. the CER and the CLR.

Cell errors and cell loss can occur due to bit errors in the ATM transmission systems (physical layer). For the sake of simplicity, it is assumed that the bit errors occur independently. BER_h is defined as the probability that an error occurs in the cell header and BER_i as the probability that an error occurs in the cell information field, so $BER_h \approx 5 \times 8 \times BER$ and $BER_i \approx 48 \times 8 \times BER$.

The CER at ATM level is the probability that the information field in the ATM cell is incorrect. If $BER_i \ll 1$, then $CER = 1 - (1 - BER_i)^{48x8} \approx 384\ BER_i$.
Cell Loss at the ATM layer occurs due to two causes:

- Cell loss due to bit errors in the header taking Header Error Control (HEC) into account.
- Cell loss due to buffer overflow in network queues.

The Header Error Control (HEC) covers the entire cell header. The code used for this function is capable of single-bit error correction or multiple-bit error detection.

Cell loss because of buffer overflow is ignored, since this is not dependent on bit errors. The discarded cell probability and the probability of valid cells with errored headers as a function of random bit error probability (BER) is given in figure 15 [I.432]. The discarded cell probability is the probability that a cell is discarded because of an errored header, while the probability of valid cells with errored headers indicates the situation that the cell header is errored so that the HEC is accidently correct.

When the BER is 10^{-9} or less, it can be concluded that both CER and CLR due to random bit errors at the physical layer can be neglected.
Cell loss will therefore mainly be caused by buffer overflow somewhere between the transmitter and receiver.

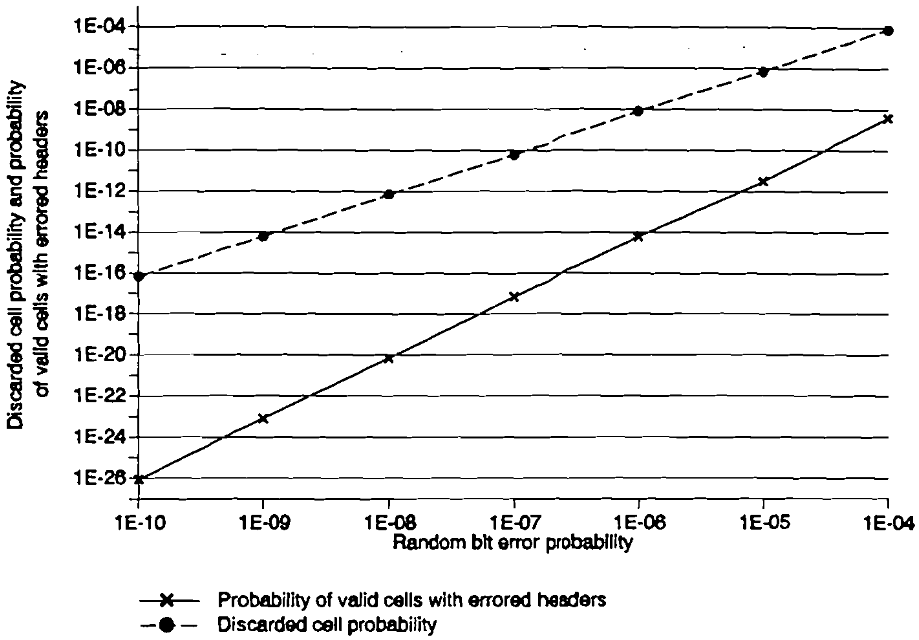
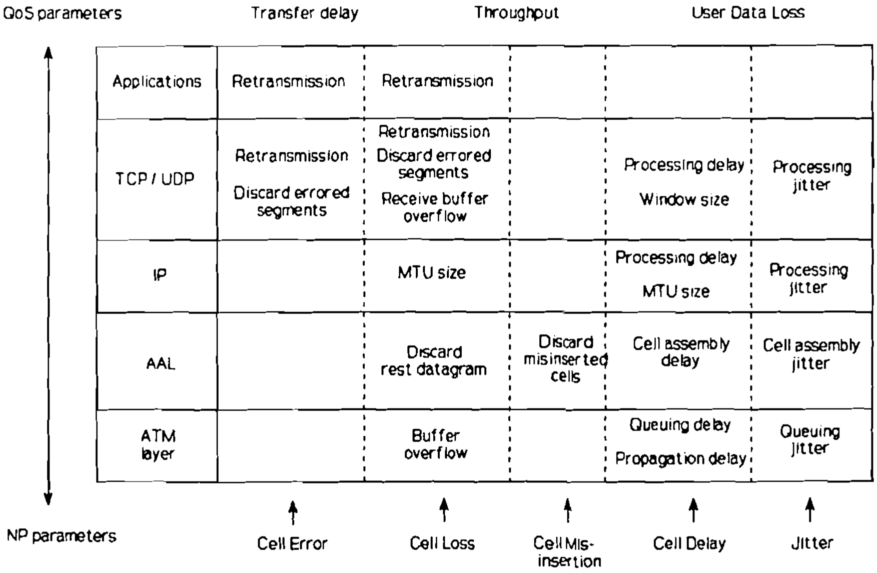


Figure 15 Influence of random bit errors on CLR and CER.

6.5 Influence of NP parameters on QoS parameters

In this paragraph the influence of NP parameters on the QoS parameters is examined. In figure 16 all operations and features that might influence the QoS are depicted. The influences of bit errors at the physical layer are not taken into account.



QoS parameters Transfer delay Throughput User Data Loss

Applications	Retransmission	Retransmission			
TCP / UDP	Retransmission Discard errored segments	Retransmission Discard errored segments Receive buffer overflow		Processing delay Window size	Processing jitter
IP		MTU size		Processing delay MTU size	Processing jitter
AAL		Discard rest datagram	Discard misinserted cells	Cell assembly delay	Cell assembly jitter
ATM layer		Buffer overflow		Queuing delay Propagation delay	Queuing jitter
NP parameters	Cell Error	Cell Loss	Cell Misinsertion	Cell Delay	Jitter

Figure 16 Layered performance model for TCP/IP and UDP/IP applications.

At the ATM layer cells mainly get lost because of buffer overflow. Cell delay is caused by queuing and propagation delays. Jitter occurs due to discrepancies in queuing delay. The amount of buffer overflow, queuing delay and queuing jitter mainly depend on the buffer size.

At the data receiver, the ATM cells are reassembled to datagrams by the AAL5. AAL5 provides a non-assured data transfer service, it is up to higher-level protocols to provide retransmissions. AAL5 performs a HEC on the cell header. In case an ATM cell header is errored, AAL5 will discard both this cell and the rest of the cells belonging to the AAL5 PDU. The cell assembly delay is the delay caused by the transference of a number of cells into an IP datagram and vice versa.

The MTU size is the most important parameter at IP level. In case of a small MTU size, the percentage of IP and TCP overhead is large and many ATM cells are needed to send an amount of User Data. This will decrease the efficiency and the throughput. The number of cells that is discarded in case of cell loss by AAL5 depends on the MTU size. Thus a large MTU size is favourable regarding the efficiency, but many cells have to be retransmitted in case cell loss occurs and AAL5 discards the total datagram. Furthermore some processing time is needed to add or distract the IP header.

When AAL5 discards an errored cell and the rest of the datagram, at TCP level no segment is received and a timeout will occur. Then the whole segment has to be retransmitted, i.e. in case of a large segment size (and thus large MTU), many cells have to be retransmitted. The TCP segment is retransmitted performing fast retransmit and fast recovery algorithms. Besides the time needed to retransmit a segment, the QoS is also decreased by the congestion avoidance or slow start algorithm, i.e. the send window is decreased. TCP also performs an end-to-end checksum, and errored segments are discarded. Since AAL5 already discards the datagram in case of cell loss and the number of errored cells is neglectable, the number of errored segments will be very small.

An important parameter for flow control at TCP level is the window size. The window size is dynamically arranged by the protocol, only the maximal window size can be set (see paragraph 5.5.2).

In case of UDP applications no end-to-end checksum, retransmissions and flowcontrol is performed at the network layer. The application should initiate retransmissions in case of errored data or not.

6.6 Efficiency and throughput of TCP/IP over ATM

An interesting question is how (in)efficient TCP/IP applications over ATM are. Efficiency is defined as the average fraction of the cell's 53 bytes that are actually used to carry User Data. The translation of IP datagrams into fixed sized ATM cells leads to a potentially inefficient use of bandwidth, because cells are only totally filled in case the datagram size is a multiple of 48, which is depicted in figure 17. In this figure, the inefficiency due to IP and TCP headers is not taken into account.

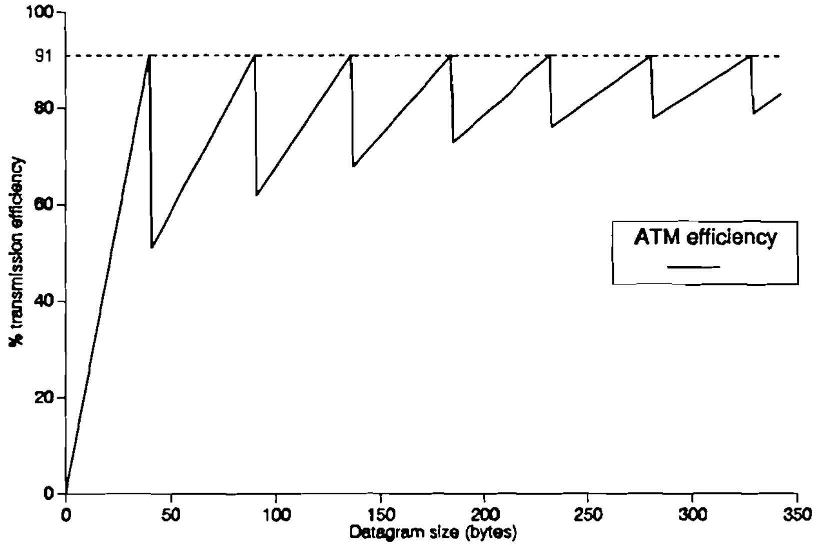


Figure 17 *ATM efficiency.*

The efficiency is not the same as the throughput, even though there is a relationship. The throughput is defined as the time needed to transfer a certain amount of User Data, so the throughput is expressed as a number of bits per second. When no cell loss occurs, a high efficiency implies a high throughput, since a minimal number of cells is needed for data transfer.

However, in case cell loss is considered, high efficiency and low throughput can go together. In case of a large MTU, the ATM cells will contain little IP or TCP overhead, so the efficiency is high, but cell loss occurs many cells have to be retransmitted, and the throughput will decrease. So the MTU influences the sensitivity for cell loss, but also the efficiency of bandwidth use and the throughput.

Both the efficiency and the throughput are measured at the receiver. So transmitted, but discarded cells are not taken into account. Only the cells and segments received correctly are used to calculate the efficiency and the throughput.

In the following examples the theoretical efficiency and throughput is calculated for a situation without and with cell loss:

■ Example 3

An amount of 10^5 bytes of User Data is transmitted over a 10^6 bit link using TCP/IP and the MTU is 48 bytes. To send this amount of User Data, 4167 datagrams of 2 cells and 1 datagram of 1 cell (thus 8335 cells in total) are generated (this can be calculated using paragraph 7.4.2, 7.5.2). The efficiency can be calculated: 8335 cells are needed to send 10^5 bytes, i.e. 12.00 information bytes per cell. The efficiency is 12.00/53 = 22.64 %

The time needed to transfer 8335 cells (cell loss, queuing delay and other delays not taken into account) is (8335 x 53 x 8)/10^6 = 3.53 seconds. So the throughput is (10^5)/3.53 = 2.83 10^4 bits/s.

In case the MTU is changed into 8192 bytes, while the amount of User Data remains the same. Now 12 datagrams of 171 cells and 1 datagram of 43 cells (thus 2095 cells in total) are generated. So the efficiency is 10^5/(2095 x 53) = 90.1 %. The propagation time of 2095 cells is (2095 x 53 x 8)/10^6 = 0.89 seconds, and the throughput is 1.13 10^5 bits/s.

Thus the efficiency becomes better when the MTU size is large. When no cell loss occurs, also the throughput is higher for a large MTU size.

■

■ Example 4

In example 3 no cell loss occurred, but in this example the influence of a Cell Loss Ratio (CLR) of 10^{-4} is examined. The same amount of User Data is transmitted and the same MTU sizes are used.

The probability that cell loss results in an errored datagram is:

$P_{datagram} = 1 - (1 - CLR)^n$, where n is the number of cells in a datagram.

In case of an MTU of 8192 bytes, the $P_{datagram}$ for n=171 is 1.70 10^{-2} and for n=43 $P_{datagram}$ is 4.29 10^{-3}. So averagely, 12 x 171 x 1.70 10^{-2} + 40 x 3.99 10^{-3} = 35.04 cells have to be retransmitted.

In practise only complete datagrams will get lost, while the calculation above gives the statistical average cell loss. So about four times this amount of User Data can be transmitted without cell loss, and the throughput will be identical to example 1. However, the fifth time one cell gets lost and therefore AAL5 discards the total datagram of 171 cells. Retransmitting a datagram takes time: (171 x 53 x 8)/10^6 = 7.25 10^{-2} seconds, so the throughput will decrease. The throughput is extra decreased because of the fast retransmit, fast recovery and slow start algorithms TCP performs when a segment is lost (α seconds). So the throughput is 10^5/(0.89 + 7.25 10^{-2} + α). For α is 0.10 seconds, the throughput becomes 9.41 10^4 bits/s, so due to the cell loss the throughput has decreased 16.7%. Still in total 2095 cells are received, so the efficiency has not changed.

For an MTU size of 48 bytes, most datagrams consist of two cells (n=2) and the $P_{datagram}$ = 1.999 10^{-4}, while one datagram consists of one cell (n=1) and the $P_{datagram}$ = 1 10^{-4}.

Averagely, the number of cells that have to be retransmitted is: 4167 x 2 x 1.999 10^{-4} + 1 10^{-4} = 1.67 cells. So certainly less cells need to be retransmitted than for an MTU of 8192 bytes, while the number of cells to transmit is larger for small MTU sizes.

In practise only complete datagrams are retransmitted, so the second time this amount of User Data is transferred, cell loss will occur and a total datagram (2 cells) has to be retransmitted. Again this retransmission will cause extra delay: $(2x53x8)/10^6$ = 8.48 10^{-4} seconds. The actions taken by TCP in case a segment is lost, cause extra delay (α seconds). So now the throughput is $10^5/(3.53 + 8.48\ 10^{-4} + \alpha)$. Again suppose α is 0.10 seconds, then the throughput becomes 2.75 10^4 bits/s, so the throughput only decreases 2.68% due to cell loss.
The efficiency has not changed, since still in total 8335 cells are received.

Thus, when cell loss occurs, the throughput decreases a lot for a large MTU size. In case of a small MTU size, the throughput only decreases a little.

■

The efficiency of TCP/IP over ATM is low in case small amounts of User Data are sent or User Data is sent with a small MTU size. ATM should not be blamed for this, TCP/IP causes the low efficiency.

7 ATM network experiments

7.1 Introduction

In the previous chapters a theoretical introduction is given about network performance of an ATM network, Quality of Service, the usefulness of statistical multiplexing and TCP/IP over ATM. In the next chapters a number of ATM network experiments is performed. The objectives of these ATM network experiments are to gain experience with ATM equipment, ATM traffic control parameters and to examine the influence of several settings and parameters on the performance of data transport (TCP/IP applications) over ATM.

The network experiments can be divided into three parts:

- Experiments to examine the performance of a connection between a single source and a single destination.
- Multiplex experiments using two or more sources and a single destination. The traffic flows are generated ourselves, so that the traffic parameters are predictable.
- Multiplex experiments using applications in practise, i.e. unpredictable traffic flows. For this experiment the results of the other tests will be used.

In this chapter only the performance of the network without multiplexing is considered. Multiplexing a number of sources over an ATM link, using predictable traffic flows is performed in chapter 7. After the results of these tests are analyzed, in future the third part can be performed, using a large number of applications in practise. However, these multiplex experiments are outside the scope of this report.

Next, some information is given about the ATM Pilot and test equipment used to perform the network experiments. Then a number of network experiments is described and the results are depicted and analyzed. Either NP parameters, traffic control parameters or parameters at IP or TCP level are varied, while the performance is measured.

Topics that are considered are:

- The segmentation of User Data into UDP datagrams or TCP segments and the translation into ATM cells.
- The performance differences between UDP/IP and TCP/IP over ATM.
- The influence of the MTU size on the efficiency and throughput of UDP/IP and TCP/IP over ATM.
- The influence of peak rate limitation on the throughput of UDP/IP and TCP/IP over ATM.
- The influence of cell loss on the throughput of TCP/IP over ATM.
- Influence of extra traffic parameters on the throughput of TCP/IP over ATM.

7.2 ATM pilot

An important reason for the acceleration of the ATM development in Europe has been the decision of 17 European operators, among which Royal PTT Nederland, to carry out a European ATM pilot during 1994/1995. The main goals of this pilot are to achieve experience in running an operational (international) ATM network and to test ATM services in practice. For this reason real users will participate in the pilot. The pilot will provide the opportunity to obtain the experience needed to verify and complete the international B-ISDN/ATM standards provided by the ATM Forum and the ITU. In figure 18, a configuration of the European Pilot is depicted [Vries94].

Most operators have started a national pilot in parallel to the European pilot. On July 1st 1994, PTT Telecom NL started a national ATM pilot in cooperation with SURFnet in which different broadband applications are tested in practice. This project will run until the end of 1995. During the project a variety of new broadband services and applications will be developed. The current operational and planned configuration of the pilot network is shown in figure 19.

SURFnet bv. was the first customer for the Dutch national ATM pilot. This organisation supplies advanced telematic services to the higher education and research community in the Netherlands. It represents a large and innovative group of (non-business) users. In order to deliver a new network for high speed communication services, a partnership was started with PTT Telecom in December 1993. The pilot network of PTT Telecom will be used for this purpose. At the first stage, ATM switches were provided by PTT Telecom for two university locations in the netherlands (SARA/Amsterdam and ACCU/Utrecht), see figure 19. During the second stage in 1995, seven campus networks of other universities and research institutes were connected to the network as well.

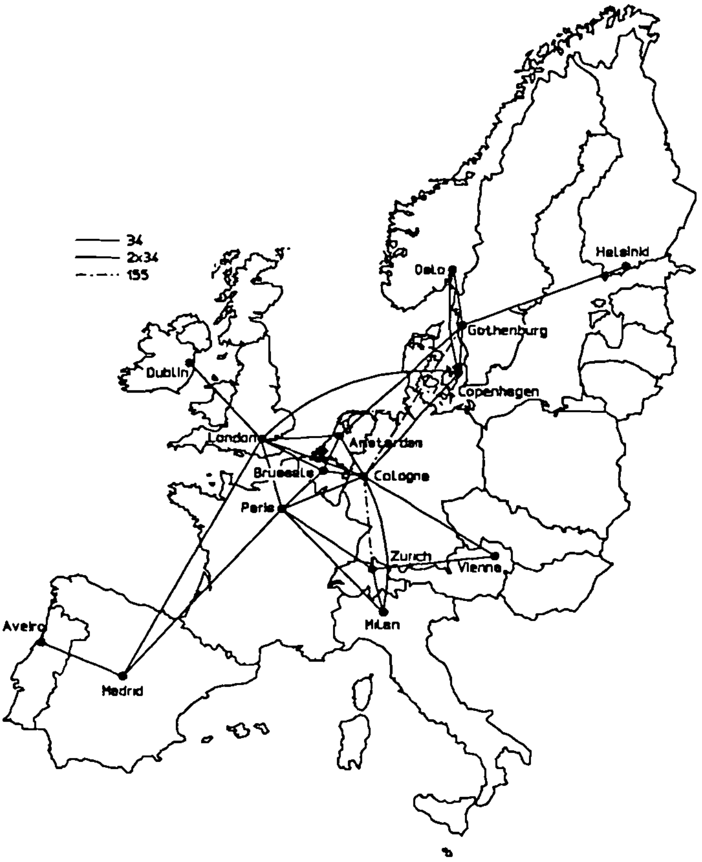


Figure 18 Configuration of the European Pilot.

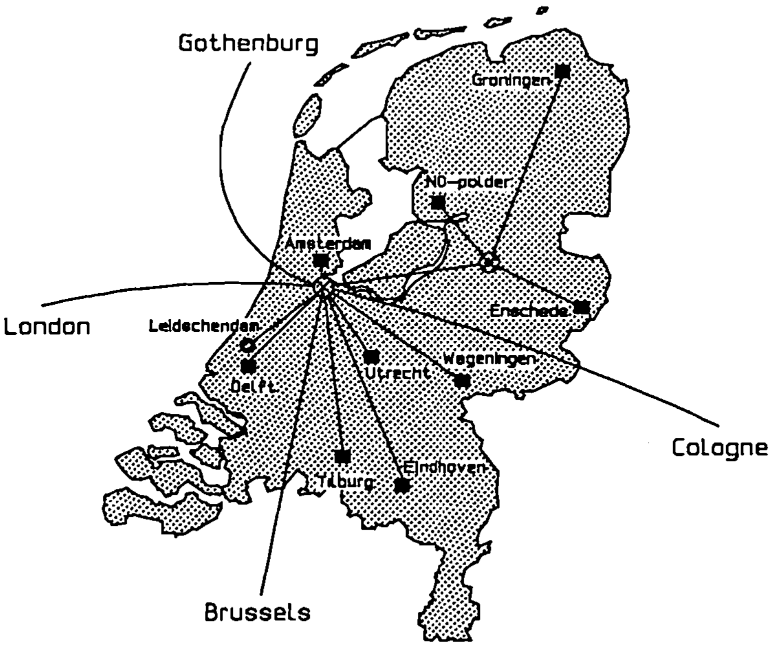


Figure 19 Configuration of the national ATM pilot network.

SURFnet plans to test the following applications and services during this project: interactive visualisation of medical images, (desktop) video conferencing, PABX interconnection, computational physics, and interactive consultation of medical experts. Furthermore, various protocol implementations will be supported as IP over ATM, IP over Ethernet, and IP over Frame Relay.

Another ATM pilot customer is the European Design Centre (EDC) in Eindhoven. This high tech company provides CAD/CAM processing facilities to its customers. Using broadband, EDC's customers can have real time access to EDC's CAD/CAM systems, considerably speeding up the design and manufacturing process and avoiding the cost-ineffective investment in powerful CAD/CAM systems.

Several organisations, both network providers and customers, are interested in the performance of the operational ATM pilot. For the performance experiments equipment and facilities of the ATM pilot are used, because this pilot provides the opportunity to test the influence of certain parameters, traffic control functions and multiplexing on the NP and QoS of real applications in practise.

7.3 Experiment equipment

7.3.1 Experiment configuration

In the experiment configuration, depicted in figure 20, four data sources and/or receivers are available, i.e. two EMMA Sun SPARCstations and two Silicon Graphics Indy workstations.

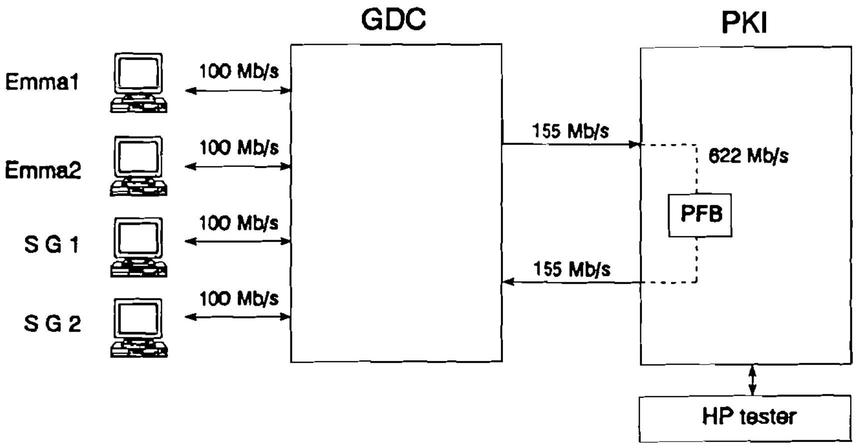


Figure 20 Experiment configuration.

All workstations are connected to the GDC switch that will be deepened out in the next paragraph. In order to perform measurements at ATM cell level, the HP tester or the Police Function Board (PFB) can be used. This board is not installed in the GDC switch, but in the PKI switch. Also the HP tester is only connected to the PKI switch. The GDC switch is connected to the PKI switch via a 155 Mb/s link.

7.3.2 FORE GIA-100 ATM Adapter

In order to connect the Silicon Graphics workstations to the ATM network, FORE GIA-100 ATM Adapter cards are included. The FORE 100-series is the first generation of ATM terminal adapters supplied by FORE Systems. The FORE GIA-100 ATM Adapter translates information of TCP/IP and OSI protocols to ATM cells via AAL3/4 and AAL5 and is equipped with a 100 Mb/s TAXI interface.

7.3.3 FORE SBA-200 ATM Adapter

The FORE 200-series is the second generation of ATM terminal adapters supplied by FORE Systems. In the EMMA workstations the FORE SBA-200 ATM Adapter is installed, that provides both AAL3/4 and AAL5 functions and features a peak rate limitation function. The adapter cards use 100 Mb/s TAXI on their external ports, providing cell based transmission at a cell rate of up to 235,849 cell/s.

7.3.4 GDC APEX switch

The switch used for the experiments is called General DataComm APEX switch (GDC switch) [GDC94], [Dijk95]. The GDC APEX switch has been designed as an access or local switch supporting many different types of interfaces towards the customer. The switch consists of a central switch fabric surrounded with interface modules, as shown in figure 20, 21.

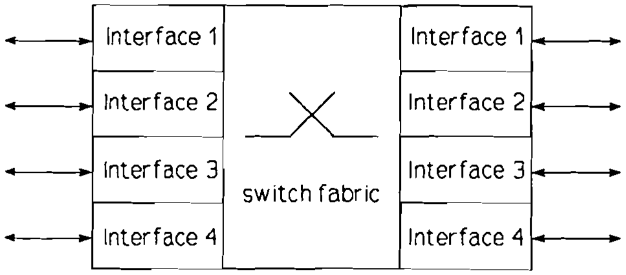


Figure 21 *General concept of the GDC APEX switch.*

The switch fabric routes ATM cells from the interface at the input side towards the appropriate interface at the output side. The switch fabric in the APEX used in the experiments has a throughput of 3.2 Gb/s. The number of interfaces and the amount of traffic offered by each interface is limited. In the interfaces the relevant transmission conversions are made in both directions. In the incoming direction, an input buffer is included to accomplish rate adaption between the interface speed and the switch fabric. The size of the input buffer is not specified.

As the switch fabric is barely loaded, these input buffers are not used and have no effect on the results of the experiments. The number of cells arriving at the interface is counted. This number is accessible via the interface control function (GDC Manager). Also policing (UPC function) is implemented in the GDC switch, but this policing function could not be used in the experiments.

In the outgoing direction, from switch fabric towards the interface, output buffering is performed to allow for rate adaption between the high-speed switch fabric and the output rate of the interface. Though the software and originally available documentation suggested otherwise, the C-series cards are equipped with only a single output buffer with a capacity of 31 cells. The number of cells that exit at the interface is counted. This number is accessible via the interface control function (GDC Manager).

The originally available documentation [GDC94] suggested the presence of H-series cards and a dual output buffer. CBR and high priority VBR traffic is placed into the high priority 63 cell deep output buffer, while medium priority and Best Effort (low priority) traffic is placed into an larger 1171 cell deep output buffer. Since some experiments reported unexpected cell loss, the size of the output buffer has been verified (see Annex D). For the experiments performed in these report, the GDC switch has been equipped with C-series cards and thus a 31 cell deep output buffer. For future experiments this might be improved.

7.4 Performance of UDP/IP over ATM

7.4.1 Introduction

To gain experience with the equipment, the UNIX-commands and monitor utilities data was first sent from EMMA2 to EMMA1 (VP=0, VC=32) via the GDC switch, as depicted in figure 22.

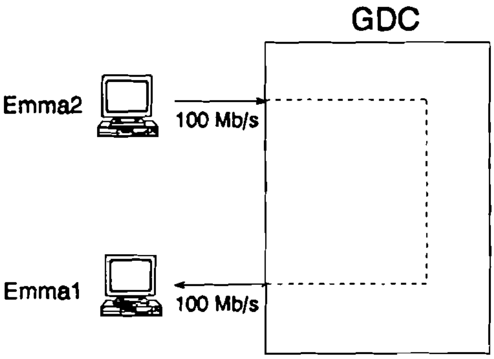


***Figure 22** Data transport experiment.*

Data is sent using a C program called udpt at EMMA1 and received by using udpr at EMMA2. Information is sent in UDP/IP packets and the following parameters can be specified in udpt:

- The burst size 's' (in bytes)
- The interval between the bursts '*t*' (in micro-seconds)
- The number of bursts '*n*' to be transmitted (an integer)

These UDP/IP packets are translated into ATM cells by the AAL5 layer. This is depicted in figure 23.

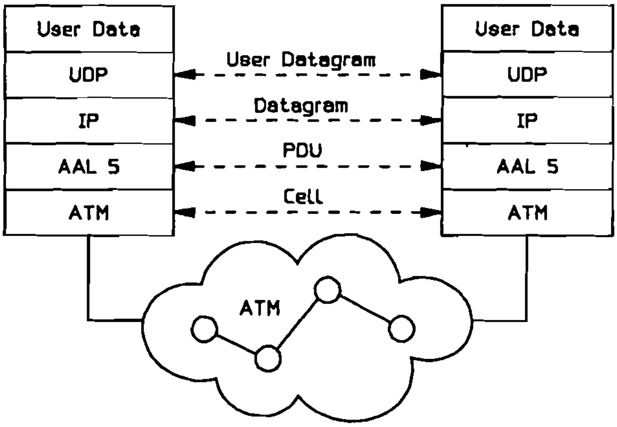


***Figure 23** Data transport via UDP/IP over ATM.*

7.4.2 Translation of UDP datagrams into ATM cells

For the first experiment, the interval time between the burst is first set to 1 second and 100 bursts are transmitted, while the burst size is varied. The MTU size is maximal (default 9188 bytes [RFC1577]), so all data fits into one UDP datagram. This way no attention has to be paid to the segmentation of data files into UDP datagrams.

The goal is to examine how the burst size relates to the number of ATM cells that are transmitted and whether this is conform the theory.
The results are given in the table 5.

■ *Table 5 Translation of UDP datagrams into ATM cells.*

Data size (bytes)	*Cells (#)*	*datagrams (#)*
12	*1*	*1*
13	*2*	*1*
1020	*22*	*1*
1021	*23*	*1*
2048	*44*	*1*
8172	*171*	*1*
8173	*172*	*1*
8192	*172*	*1*
9000	*189*	*1*
9001	*0*	*0*
10000	*0*	*0*

It can be concluded that 36 bytes of overhead are added to the bursts and the total number of bytes is put into cells with 48 information bytes. In case of a burst of 8172 bytes, for example, the number of bytes overhead included is 8208, which results in 8208/48 = 171 cells. When a larger burst (8173 bytes) is transmitted, this results in 172 cells. A total overhead of 36 bytes is conform the theory, since UDP adds an 8 bytes trailer [I.363]. The burst size is limited to 9000 bytes; larger bytes are not transmitted at all.

7.4.3 Segmentation of files into UDP datagrams

During this second experiment the segmentation of files into UDP datagrams and ATM cells is examined. If the amount of data is smaller than the selected MTU size, the AAL receives 28 bytes (20 bytes IP header + 8 bytes UDP header) plus the selected amount of data (see previous experiment). Otherwise, the file is segmented into several smaller UDP datagrams. This process will be examined in detail.

The maximum MTU size was set to 48 bytes, using the `atmconfig` utility. A small MTU size forces the IP layer to fragment the IP datagram into many small IP packets. The utility `atmstat` was used to monitor the number of cells and UDP datagrams sent and received. The command `./udpt emma2 -n 1000 -t 1000000 -s #` was used with different values for the size parameter. The table below lists the number of cells and UDP datagrams that resulted from the different amounts of data.

■ ***Table 6*** *Segmentation of files into UDP datagrams.*

Data size (bytes)	*Cells (#)*	*UDP datagrams (#)*		*Data size (bytes)*	*Cell (#)*	*UDP datagrams (#)*
12	*1*	*1*		*89*	*9*	*5*
13	*2*	*1*		*136*	*12*	*6*
20	*2*	*1*		*137*	*13*	*7*
21	*3*	*2*		*208*	*18*	*9*
40	*4*	*2*		*209*	*19*	*10*
41	*5*	*3*		*1048*	*87*	*44*
64	*6*	*3*		*1049*	*88*	*45*
65	*7*	*4*		*8192*	*683*	*342*
88	*8*	*4*		*9000*	*751*	*376*

The number of cells and UDP datagrams can be explained and are conform the theory. A MTU size of 48 bytes leaves 20 bytes open for data (8 bytes UDP header and 20 bytes IP header), however, if an UDP datagram is segmented, its data portion must be broken on 8 octet boundaries, according to [RFC791]. The last datagram can be filled until the MTU size according to [RFC791], but the Berkeley implementation (Sun TCP/IP sources) works differently and also breaks the last datagram on an 8 bytes boundary. Then an 8 bytes AAL trailer is added to the UDP datagram and it is segmented into ATM cells.

So a data size of 12 bytes gives one UDP datagram of 40 bytes and one completely filled cell. Two cells and one UDP datagram are needed for a data size of 13 bytes. An UDP datagram is completely filled for a 20 bytes data size, so for a 21 bytes data size two UDP datagrams are necessary. In that case the first UDP datagram contains 8 bytes UDP header, 20 bytes IP header and 16 bytes of data which gives 2 ATM cells and the second UDP datagram contains 20 bytes IP header and 5 bytes data, which results in 1 cell. In total there are 2 UDP datagrams and 3 ATM cells, as shown in the next figure.

For a 64 bytes data size, 3 UDP datagrams are needed; the first datagram is filled with 28 bytes overhead and 16 bytes data, which results in 2 cells and the second and third datagram contain 20 bytes overhead and 24 bytes data, which both result in 2 cells also. So in total there are 6 ATM cells. For every next 24 databytes a new UDP datagram is needed. A UDP datagram is translated into one cell in case it contains 8 or 16 databytes and two cells when 24 databytes are included.

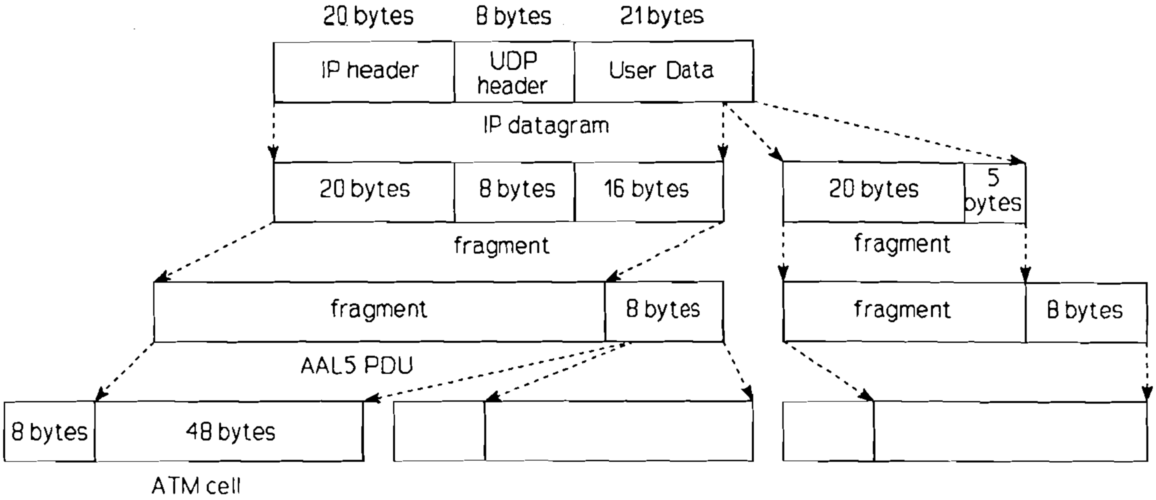


Figure 24 *IP fragmentation.*

Because of the small MTU size the efficiency is very low; to send 9000 bytes, 751 cells are needed, while for an MTU size of 8192 bytes only 189 cells are needed to send 9000 data bytes. The small MTU size was only set to explain the process of data segmentation into UDP datagrams.

7.4.4 Throughput of UDP/IP over ATM

In order to examine the throughput of UDP/IP over ATM, a connection is set up between EMMA1 and EMMA2 (VP=0 and VC=32) and the burst size is taken large, while the interval time will be very small (or zero). The following command is given at EMMA2: `./udpt -n 30000 -s 8192 -t 0 emma1`, so 234.38 MB are transmitted to EMMA1. The experiment is repeated for the opposite direction, i.e. from EMMA1 to EMMA2. The peak rate is unrestricted and the MTU size is varied.
Using a watch, the time to transmit this information is measured and the throughput of UDP/IP over ATM is calculated. The throughput is defined as the file size divided by the time needed to transmit this file. The results are given in table 7 and figure 25.

***Table** 7 Throughput and efficiency of UDP/IP over ATM.*

MTU	*# cells*	*# UDP datagrams*	*Efficiency (%)*	*Transmit time (s)*	*Throughput (Mb/s)*	*Transmit time (s)*	*Throughput (Mb/s)*
For both EMMAs				*EMMA2 → EMMA1*		*EMMA1 → EMMA2*	
9188	*5,160,000*	*30,000*	*89.9*	*24*	*85.9*	*33*	*62.5*
8192	*5,190,000*	*30,000*	*89.3*	*28*	*73.6*	*38*	*54.3*
4096	*5,220,000*	*90,000*	*88.8*	*30*	*68.7*	*41.5*	*49.7*
2048	*5,250,000*	*150,000*	*88.3*	*37*	*55.7*	*50*	*41.2*
1024	*5,430,000*	*270,000*	*85.4*	*48*	*43.0*	*64*	*32.2*
512	*5,550,000*	*510,000*	*83.5*	*70*	*29.5*	*87*	*23.7*

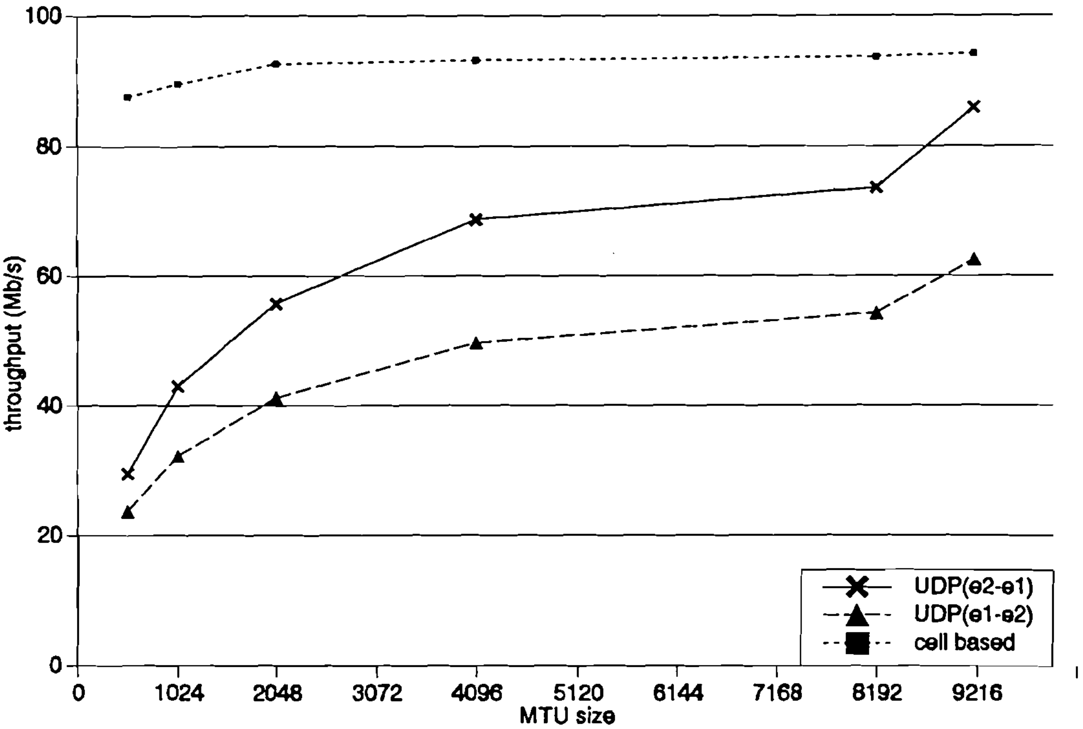


***Figure 25** Throughput of UDP/IP over ATM versus MTU size.*

It should be noted that the time measurements are done by hand and therefore not very accurate. As can be seen in figure 25, the throughput of UDP/IP over ATM decreases a lot when the MTU size is decreased. Apparently the throughput of UDP/IP over ATM is lower for data sent from EMMA1 to EMMA2 than for the opposite direction. A reason for the throughput difference might be the fact that EMMA2 is a SunSPARC 20 workstation, while EMMA1 is only a (less powerful) SunSPARC 10 workstation.

When the MTU size becomes smaller, the percentage of overhead becomes larger, more cells are needed to transmit the same amount of data and the efficiency becomes smaller. Since more cells have to be transmitted, the throughput is expected to decrease. The throughput based on the number of ATM cells assuming a workstation transmits cells with 100 Mb/s, is also depicted in figure 25 as 'cell based' throughput. However, the measured throughput decreases more than is expected on basis of the extra number of cells.

A reason for this might be the workstation's processing delay. The processing delay results in an interdatagram gap, which is the time needed by the workstation to generate a UDP datagram, and an intercell gap, which is the time needed to translate a UDP datagram into ATM cells. Using the measurements listed in table 7, the interdatagram gap and the intercell gap are calculated for both EMMA1 and EMMA2. This calculation is thoroughly described in Annex A. The intercell gap is respectively 1.7 µs for EMMA1 and 63 ns for EMMA2, while the interdatagram gap is 110 µs for EMMA1 and 90 µs for EMMA2. This processing delay explains the curves of figure 25. In figure 26 the throughput measurements are depicted, including the calculated throughput taking the processing delay into account.

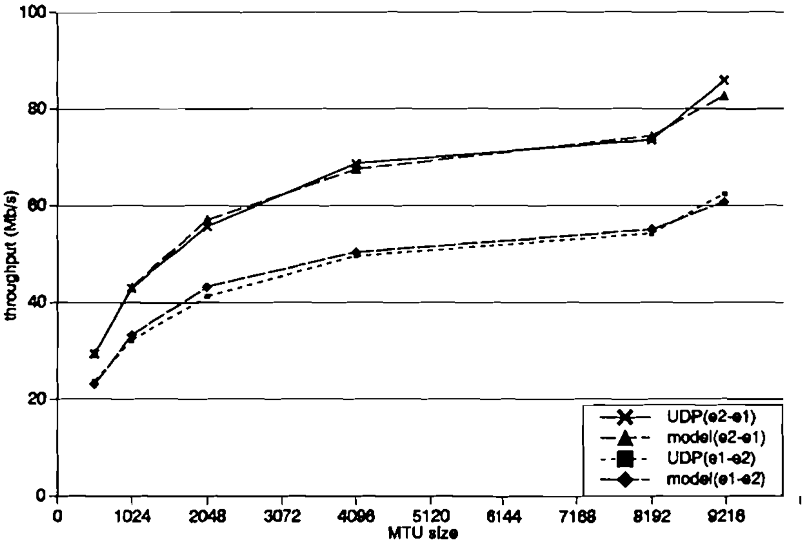


Figure 26 Calculated throughput taking the processing time into account.

7.4.5 Throughput of UDP/IP over ATM versus the peak rate

The consequences of peak rate settings at the EMMA workstations for the throughput of UDP/IP over ATM is examined, using the same experiment configuration as before. Using the command `udpt -n 50000 -t 0 -s 8192 emma1`, 3.28 10^9 bits (50000 x 8192 x 8) of User Data is transferred from EMMA2 to EMMA1 and vice versa, while the peak rate is varied between 87.3 Mb and 1 Mb. The consequences for the throughput are depicted in table 8 and figure 27.

■ Table *8 The throughput of UDP/IP over ATM versus peak rate.*

peak rate (Mb/s)	*# cells*	*# datagrams*	*time (s)*	*throughput (Mb/s)*
87.3	*8,600,000*	*50000*	*40*	*81.92*
80.0	*8,600,000*	*50000*	*40*	*81.92*
70.0	*5,911,812*	*34371*	*35*	*64.36*
60.0	*4,610,288*	*26804*	*31.5*	*55.77*
50.0	*3,305,496*	*19218*	*28*	*44.98*
40.0	*2,647,252*	*15391*	*26*	*38.79*
30.0	*1,723,440*	*10020*	*24*	*27.36*
20.0	*2,108,720*[*]	*12260*	*45*	*17.85*
10.0	*981,948*[*]	*5709*	*40*	*9.354*
5.0	*474,032*[*]	*2756*	*40*	*4.515*
2.0	*205,024*[*]	*1192*	*41*	*1.905*
1.0	*102,856*[*]	*598*	*41*	*0.956*

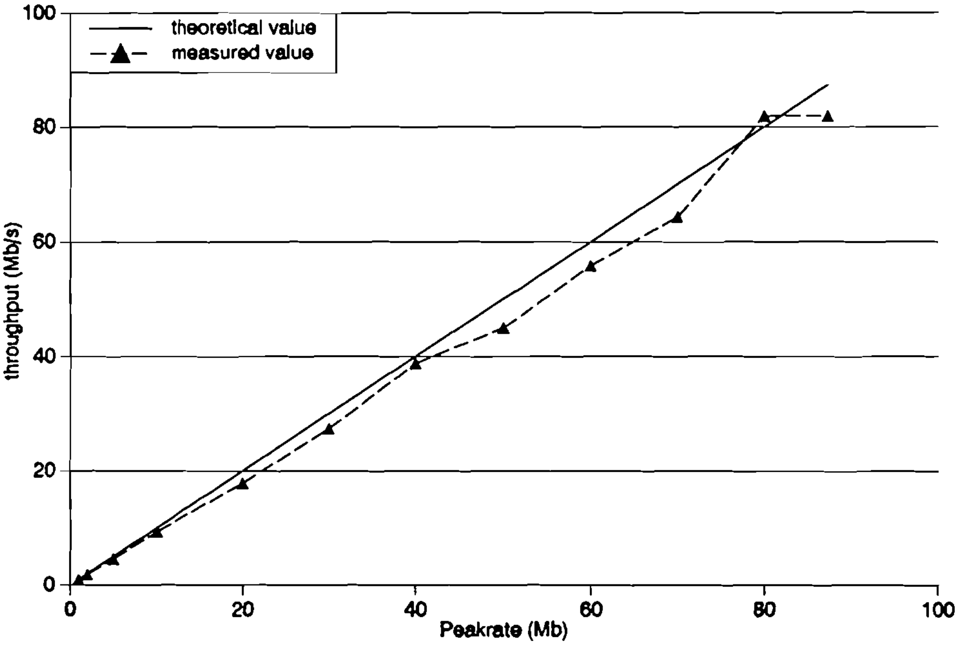


Figure 27 *The throughput of UDP/IP over ATM versus the peak rate.*

[*] *100000 datagrams of 8192 bytes are generated, in order to increase the transfer time and to increase the accuracy of the throughput measurements.*

The first issue to be noticed in table 8, is the unexpected decrease of the number of received datagrams and cells when the peak rate is reduced, according to monitor utility atmstat. However, no cell drops are reported and all transmitted cells are received at EMMA1. Apparently, the User Data has been lost before the ATM cells are transmitted, and monitored by atmstat.
At the EMMA2 SunSPARC20 workstation, the information generation program udpt will produce 50000 datagrams as fast as possible. In case of a peak rate of 80 Mb or more, this amount of User Data can be transmitted immediately, but when the peak rate is reduced, the FORE ATM card has to buffer the User Data temporarily in order to meet the peak rate setting. When this buffer is exceeded, User Data is lost even before atmstat can monitor it. In order to solve this problem, it is desirable to implement a control signal that notifies the source that the buffer is full and the data generation should be stopped temporarily.

The throughput, listed in table 8, is calculated taking only the transmitted User Data into account, i.e. (# datagrams x 8192 x 8)/time. Figure 27 shows that the measured throughput is conform the set peak rate. Apparently, the maximal throughput is 81.92 Mb/s, when the start up and finish period are taken into account. Occasionally, at a smaller time scale, a higher throughput can be reached.

In case this experiment is repeated with EMMA1 as source and EMMA2 as receiver, cell drops are displayed by atmstat at EMMA2 for a peak rate set lower than 80 Mb. For a lower peak rate all cells are dropped according to atmstat. Therefore no throughput measurements are performed for this situation. The exact reason for the cell drops is unknown and for further study.

7.4.6 Maximal throughput

For an MTU size of 8192 and maximal peak rate occasionally between 230,000 and 235,000 cells/s are received by EMMA1, not taking the start up and finish period into account. This corresponds to a capacity of 97.5 to 99.6 Mb/s. So occasionally the link capacity can almost completely be filled.

When information is sent from EMMA1 to EMMA2 using the same settings, however, occasionally only around 167,000 cells/s are received by EMMA2, which is 70.8 Mb/s. So for some reason the link capacity can not be filled by EMMA1. The MTU size and the peak rate were equal for both experiments. A reason for the throughput difference might be the fact that EMMA2 is a SunSPARC 20 workstation, while EMMA1 is only a SunSPARC 10 workstation.

7.5 Performance of TCP/IP over ATM

7.5.1 Introduction

Using the same experiment configuration as in figure 22, the performance of TCP/IP over ATM is measured. Information is sent in TCP/IP packets using a program called `ttcp -t`, which contains the following parameters:

- The length of buffers written to network '*l*' (in bytes); default 1024
- The number of buffers to send '*n*' (an integer); default 1024

Information is received using `ttcp -r`, which contain similar parameters:

- The length of the network read buffer '*l*' (in bytes); default 1024
- The number of buffers to receive '*n*' (an integer); default 1024
- Sink (discard) all data ('*s*')

An amount of *n* x *l* bytes is transmitted in TCP/IP segments as fast as possible depending on the MTU size, peak rate and Cell Loss Ratio (CLR). The TCP/IP packets are translated into ATM cells by the AAL5 layer. This is depicted in the following figure.

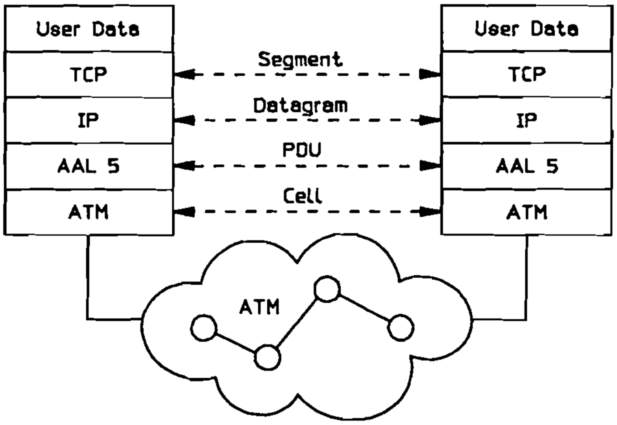


Figure 28 *Data transport via TCP/IP over ATM.*

7.5.2 Translation of User Data into ATM cells for TCP/IP

The goal of the next experiment is to understand the segmentation of an amount of User Data into IP datagrams and the translation of these IP datagrams into ATM cells in case of TCP/IP data transport.

A data file of 30 10^6 bytes is sent from EMMA2 to EMMA1 using TCP/IP. The MTU size is varied, while peak rate is unrestricted and no cell loss is introduced. The average number of cells that is measured for a certain MTU size is depicted in table 9. All measurements were repeated twice.

■ ***Table 9** Expected and measured number of cells corresponding to the MTU size.*

MTU	*Expected # cells*	*Measured # cells*	*difference*
9188	*629952*	*629771*	*-181*
8192	*629451*	*629478*	*27*
4096	*636142*	*636188*	*46*
2048	*642420*	*642710*	*290*
1024	*670736*	*670839*	*103*
512	*699149*	*701407*	*2258*

The expected number of cells have been calculated as follows:
Both TCP and IP add a header of 20 bytes to a chunk of data, so in case of an MTU size of 8192, only 8152 bytes can be occupied by data. To transmit 30 10^6 bytes, at least 3681 IP datagrams are needed. The AAL5 adds a trailer of 8 bytes, so (8192 + 8)/48 = 171 ATM cells are sent per IP datagram. In total 171 x 3681 = 629451 cells are expected. According to table 9, 629478 cells are measured, so 27 cells more are measured then expected.

According to table 9, less cells are measured than expected on basis of the calculations described before in case of the largest MTU size (9188 bytes). In fact this is strange, since the number of cells calculated is supposed to be the minimal number of cells possible, i.e. the number of cells in case all IP datagrams are filled completely. No solid explanation is found yet to explain this difference.

In case of the other MTU sizes, however, more cells are measured than calculated. Apparently not all IP datagrams were filled completely and therefore more cells were needed. Every time this experiment is repeated a different number of cells is measured, because TCP/IP is a dynamic protocol.

The difference between the calculated and measured cells is large for a MTU size of 512 bytes, but in this situation cell loss occurred, so cells had to be retransmitted (1108 drops). It is unknown yet, why cells got lost for this small MTU size.

7.5.3 Throughput of TCP/IP over ATM

The throughput of TCP/IP over ATM is measured, while parameters like the MTU size, the peak rate and the Cell Loss Ratio (CLR) are varied. Only one parameter is varied for each experiment and the other parameters are kept constant. The experiment configuration remains the same as above.

Using the command `ttcp -t -s -n600 -150000 emma2`, 30 10^6 bytes is sent from EMMA1 to EMMA2 via the GDC switch. A similar command is used to sent 30 10^6 bytes from EMMA2 to EMMA1. Parameters like the send window, receive window, the Maximum Segment Size are not taken into account, i.e. the send and receive window are set maximal (50000 bytes) and the MSS is decided by the MTU size. After all data is received, the number of bytes processed, the time needed to receive all data and the throughput is displayed.

7.5.4 Throughput of TCP/IP over ATM versus the MTU size

During this experiment the throughput of TCP/IP over ATM is measured, while the MTU size is varied between 9188 and 512 bytes. The peak rate is set to be unrestricted and no cell loss is introduced. The number of cells received and the throughput is displayed in table 10. Figure 29 shows the throughput versus the MTU size for both TCP/IP and UDP/IP over ATM.

■ ***Table 10** The throughput of TCP/IP over ATM versus the MTU size.*

MTU	*# Cells (average)*	*Throughput (Mb/s)* *EMMA2 → EMMA1*	*Throughput (Mb/s)* *EMMA1 → EMMA2*
9188	*629771*	*46.61*	*43.40*
8192	*629478*	*44.99*	*41.11*
4096	*636188*	*39.50*	*33.97*
2048	*642710*	*29.75*	*24.42*
1024	*670839*	*19.53*	*15.02*
512	*701407*	*11.16*	*11.16*

Just as during the throughput measurement of UDP/IP, depicted in the same figure, also the throughput of TCP/IP decreases more than expected on basis of the number of extra transmitted cells. It is again assumed that the interdatagram gap is the cause of the progressive decrease of the throughput when the MTU size decreases.

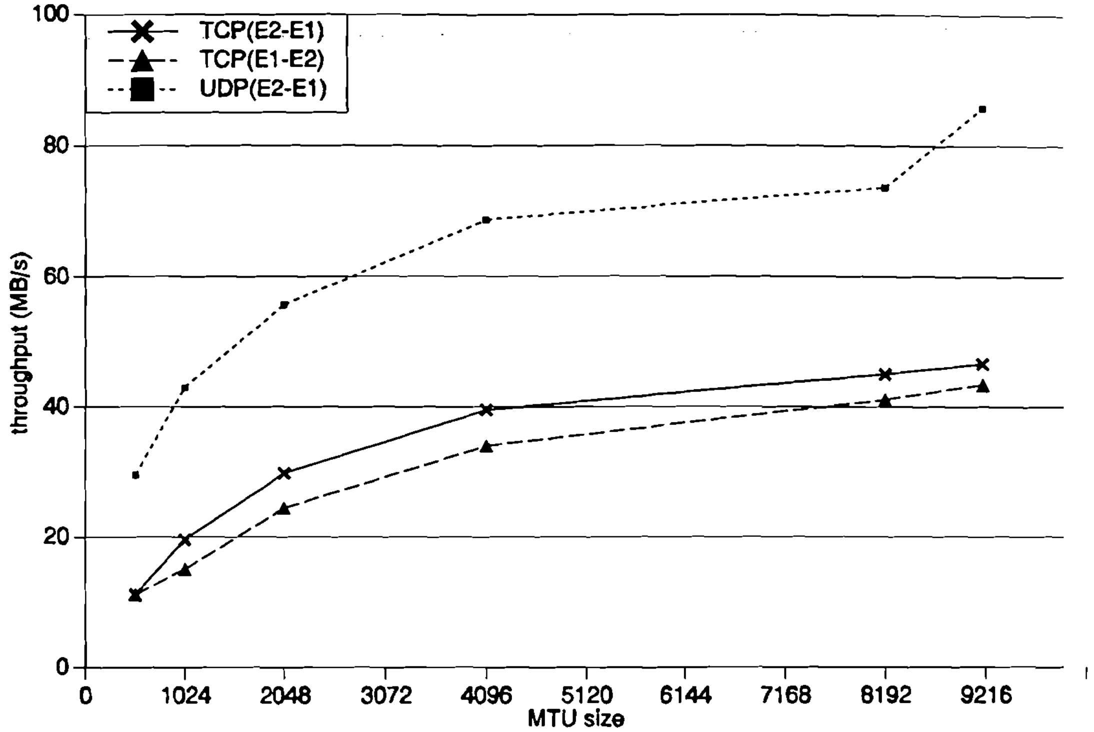


Figure 29 The throughput versus MTU size for TCP and UDP.

Secondly a difference in throughput can be seen for data sent from EMMA1 towards EMMA2 and the opposite direction. This had also already been noticed for the maximal throughput for UDP/IP traffic (see paragraph 7.4.4). Again the difference in workstation (a SunSPARC10 versus a SunSPARC20) is suspected to be the cause.

The throughput of TCP/IP over ATM is lower than the throughput for UDP/IP over ATM for all MTU sizes. TCP/IP is a connection oriented protocol, and segments have to be acknowledged before new segments are transmitted. Furthermore TCP contains mechanisms to guarantee that data is error-free, complete and in sequence. Since all this takes (processing) time, it makes sense that the throughput is lower than for UDP/IP over ATM.

Using the measurements listed in table 10, the interdatagram gap and the intercell gap can be calculated for TCP/IP over ATM. Similar calculations as described in Annex A, result in an estimated intercell gap of 3.5 μs and an interdatagram gap of 253 μs for this situation. No distinction is made between EMMA1 and EMMA2 in this situation since the difference between the measured throughput curves is small and the values of the intercell gap and interdatagram gap are just estimated. In figure 30 the throughput measurements are depicted, including the calculated throughput taking the estimated processing delay into account (model).

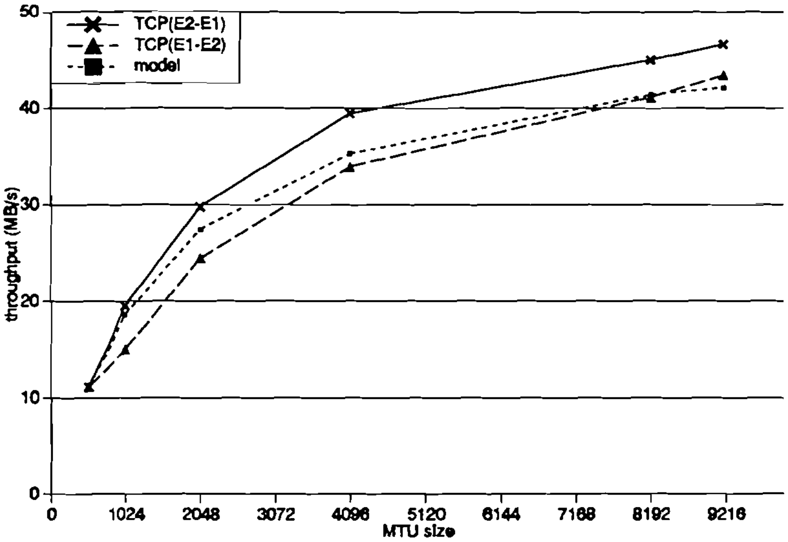


Figure 30 The measured throughput versus the estimated model.

7.5.5 Throughput of TCP/IP over ATM versus peak rate

The MTU size is set to 9188 bytes and no cell loss is introduced. Again 30 10^6 bytes is sent from EMMA1 to EMMA2 and vice versa, while the peak rate is varied between 87.3 Mb and 2 Mb. Regardless of the set peak rate about 629,662 cells and 3291 datagrams are transferred. The consequences for the throughput can be seen in table 11 and figure 31.

***Table 11** The throughput of TCP/IP over ATM versus the peak rate.*

Peak rate (Mb)	Throughput (Mb/s)		Peak rate (Mb)	Throughput (Mb/s)	
	E2→E1	E1→E2		E2→E1	E1→E2
87.3	47.35	43.40	40	37.20	35.92
80	47.35	44.02	30	27.10	26.04
70	46.88	43.40	20	18.10	17.96
60	43.40	42.82	10	9.032	9.032
55	42.82	41.12	5	4.507	3.005
50	41.12	40.06	2	1.875	1.502
45	40.42	38.59			

In order to increase the accuracy of the throughput measurements, the experiment is repeated with a larger amount of data (200 10^6 bytes).

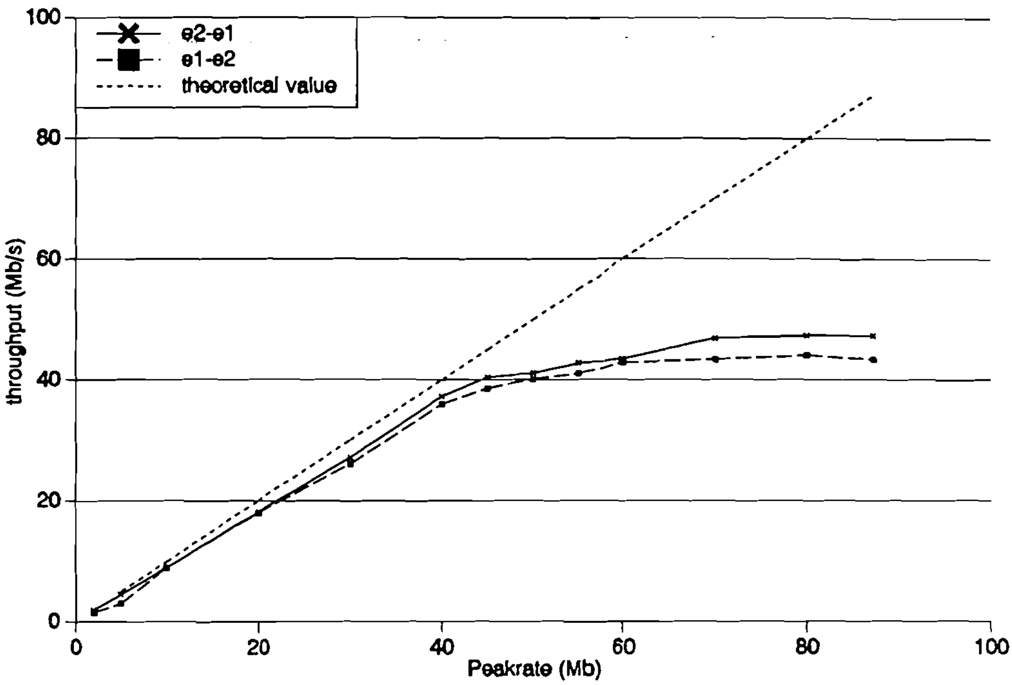


Figure 31 The throughput of TCP/IP over ATM versus the peak rate.

As expected is the number of cells and datagrams sent approximately equal for all measurements. The measured throughput is conform the set peak rate for lower peak rate values. However, for a peak rate larger than about 45 Mb, the throughput increases not much any more. Apparently, the throughput is limited by the workstations. A small difference in throughput can be seen for data sent from EMMA1 towards EMMA2 and the opposite direction. Again the difference in workstation (a SunSPARC10 versus a SunSPARC20) is suspected to be the cause.

7.5.6 Influence of Cell Loss on the throughput of TCP/IP over ATM

For this experiment the TCP/IP traffic is routed through both the GDC and the PKI switch, since the Police Function Board is connected to the PKI switch (see figure 32). An amount of 30 10^6 bytes User Data is sent from EMMA1 towards EMMA2 (VP=0, VC=32), using the command `ttcp -t -s -n600 -l50000 emma2`. The Cell Loss Ratio (CLR) is varied using the PFB (CLR is 0, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3} and 2 10^{-3}) and for each CLR the MTU is modified between 9188 and 512 bytes. The throughput and the number of cells and PDUs received and dropped are measured using the monitor utility `atmstat`. The meaning of 'cell drops' in `atmstat` is explained in Annex B. To achieve a CLR of 2 10^{-3}, the splash and Bucket Limit (BLM) at the PFB are set to 32 and 32 x 498 = 159362. Then 499 cells pass the PFB and the next cell is discarded.

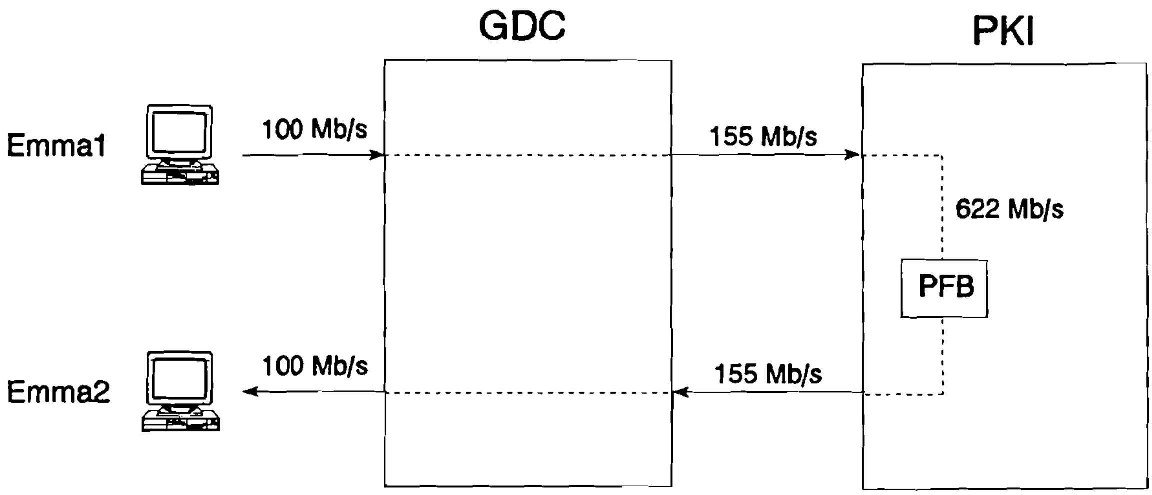


Figure 32 Cell loss experiment configuration.

The results of the experiment are depicted in figure 33 and for more details the numerical results are listed in table C.1.

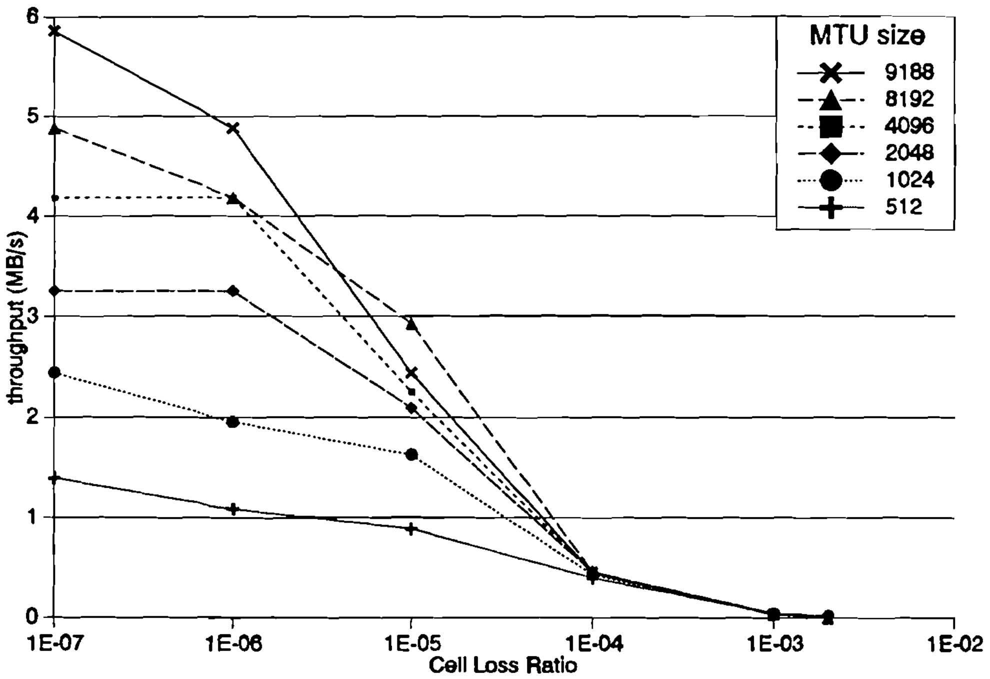


Figure 33 Throughput of TCP/IP over ATM versus Cell Loss.

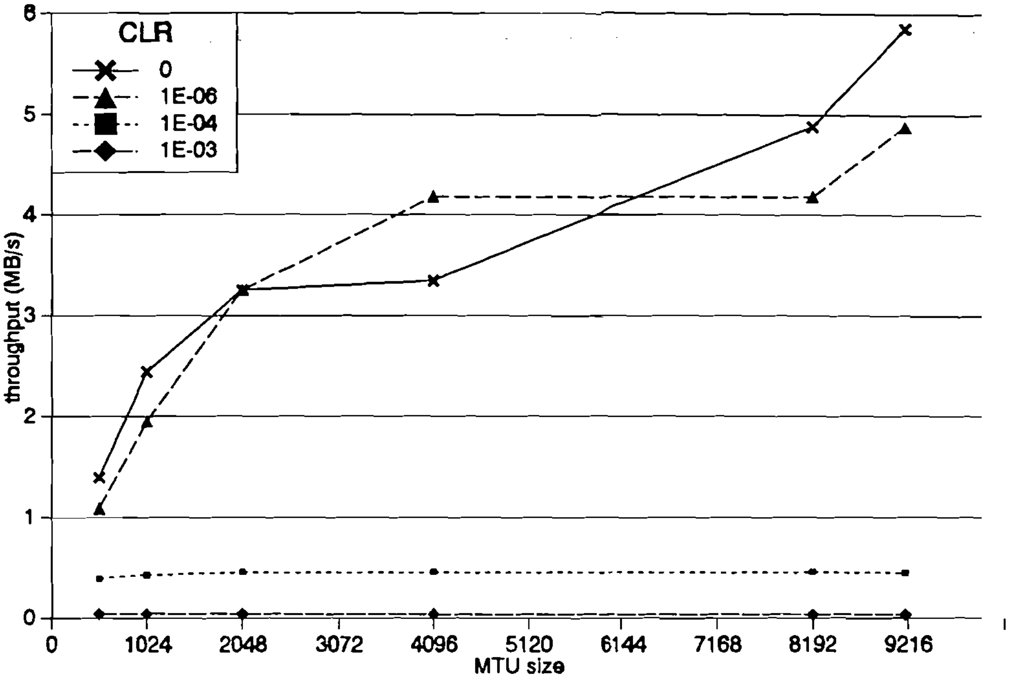


Figure 34 Throughput versus MTU for different CLRs.

For a CLR of 10^{-3} and 2 10^{-3} a file of 10^6 bytes is sent from EMMA1 towards EMMA2, using the command `ttcp -t -s -n20 -l50000 emma2`, because sending larger files takes a very long time. Therefore the number of cells differs a lot from the other measurements. As seen before, the throughput decreases rapidly when the MTU size is decreased. Secondly, the throughput also rapidly decreases when cell loss occurs.

For a CLR of 10^{-4} or more, the MTU size does not matter any more and the throughput is 3 Mb/s or less (see figure 34). For a CLR of 10^{-5} the highest throughput is reached for an MTU size of 8192 bytes in stead of 9188 bytes.

When cell loss occurs, a total segment has to be retransmitted. When segment loss is discovered, the fast retransmit and fast recovery algorithms will be performed. Using the data in table C.1, an estimation for the extra delay due to segment loss is performed. The difference in transfer delay, at measurements with identical MTU sizes but different CLR values (see table C.1), is divided by the number of retransmitted datagrams.

It is estimated that each segment loss causes an extra delay of about 0.9 to 1 second. For example, when the MTU is 9188 bytes, the calculation of the extra delay is

$$\text{extra delay} = \frac{T_{clr=10^{-4}} - T_{clr=0}}{\text{number of retransmitted segments}} = \frac{65 - 5}{65} \approx 0.92 \text{ s}$$

This extra delay due to segment loss causes the rapid decrease of throughput. When large amount of User Data has to be retransmitted, the extra transfer delay has to be taken into account. However, this is just an estimation of the extra delay due to cell loss.

In [Dijk95] the theoretical curve of the TCP/IP throughput versus the cell loss is depicted, taking an estimated extra delay due to cell loss of 1 second into account (see also figure 35). The theoretical curves complies with the measured throughput curves, so the estimated extra delay due to cell loss is correct.

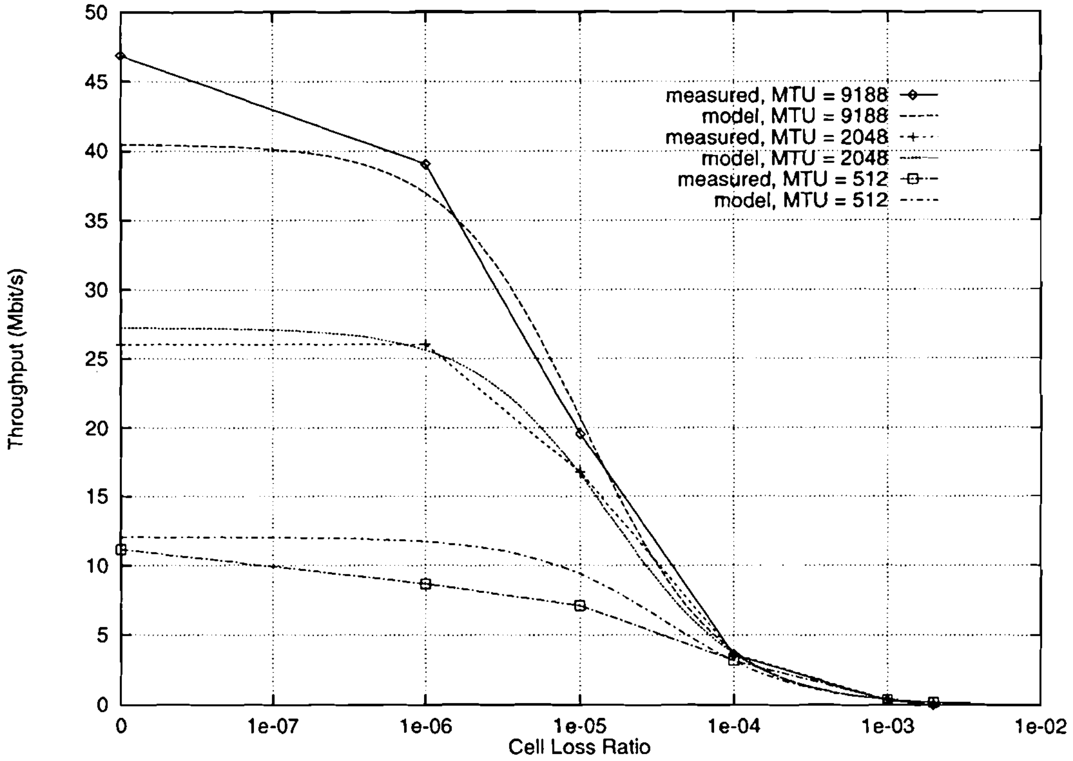


Figure 35 Measured and modeled curves of throughput versus Cell Loss.

In this paragraph, the overall (added) extra delay is just estimated. Using the HP tester the traffic stream at ATM cell level can be monitored and the separate influence of each TCP algorithm might be distinguished. This is for further study.

During this experiment the RTT was only 1 ms, but when the RTT becomes larger, it is expected that the influence of cell loss becomes even larger, since the time to recognize segment loss and the slow start or congestion avoidance period become larger.

7.6 Influence of extra traffic parameters on the throughput of TCP/IP over ATM

In the previous paragraphs the only traffic parameter that has been considered was the peak rate. In case of CBR traffic the peak rate is the only parameter, but for VBR traffic also the Sustainable Cell Rate (SCR) and the Burst Tolerance (BT) have to be considered. The influence of including the SCR and the BT in the policing scenario on the throughput of TCP/IP over ATM is examined.

Data is transferred from EMMA2 to EMMA1 via the GDC switch and the PKI switch, see figure 32. The values for PCR, SCR and BT are set at the PFB connected to the PKI switch, while at the workstations only a peak rate is set. The principle of the policing scenario, including SCR and BT is depicted in figure 36. A bucket is filled with the transfer rate (maximally the peak rate) and emptied with the SCR. The size of the bucket is BT and when this value is exceeded, information is lost.

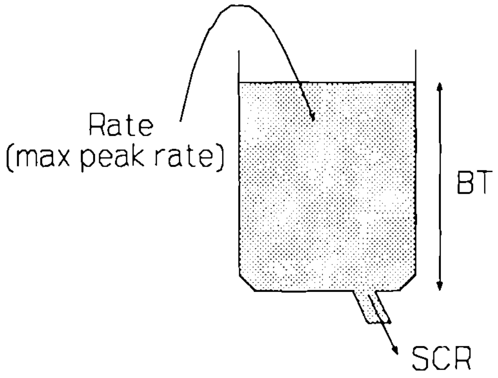


Figure 36 Policing principle using Burst Tolerance and Sustainable Cell Rate.

The source will transmit information and the TCP protocol will increase its send window every time an ACK is received. The source will send at the set peak rate until the BT is reached. Then the policing function will discard all cells until the SCR is achieved.

When cells are discarded, the source will wait for an ACK and finally time out. TCP's flow control mechanism will decrease the send window until the SCR is reached and cells can pass the PFB again, i.e. EMMA1 receives segments again. The transmitter will receive ACKs and the send window is increased until the BT is reached again. Due to the activities of TCP, it is expected that the throughput is less than the SCR. The influence of the value of the BT, MTU and peak rate compared to SCR is examined.

7.6.1 The throughput of TCP/IP over ATM versus the Burst Tolerance

The peak rate at the EMMA workstations is set at 10 Mb. At the PFB, the peak rate is also set at 10 Mb (L=4, S=225, BLM=3616) and the SCR is set at 3 Mb (L=1, S=187). The BT is varied between 400 and 12800 cells and also the MTU is changed between 512 and 9188 bytes. Using the command `ttcp -t -s -n100 -150000 emma1`, an amount of 50 10^6 bytes of User Data is transmitted from EMMA2 to EMMA1. The number of cells, datagrams, drops, the transfer delay and the throughput are measured. The results are shown in the next figures, while the numerical results are listed in table B.2. In figure 37, the throughput of TCP/IP over ATM (in Kb/s) versus the BT (in cells) is plotted for several MTU sizes. Also the throughput of TCP/IP over ATM (in kb/s) is plotted as function of the MTU size (in bytes) for several values of the BT in figure 38.

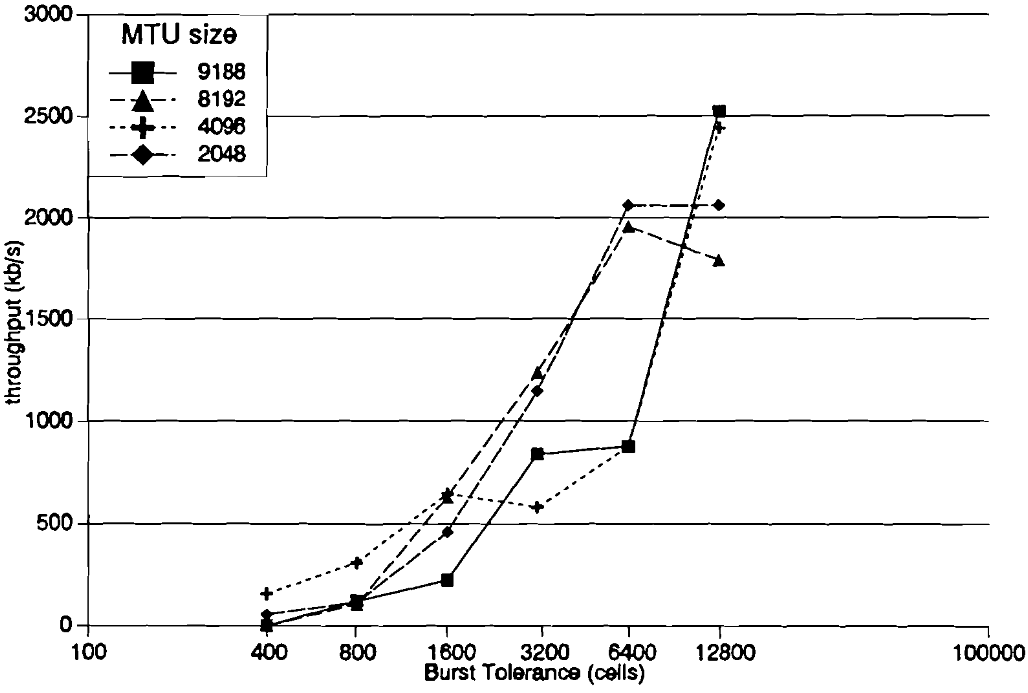


Figure 37 The throughput versus the Burst Tolerance for several MTU sizes.

It can be concluded that the throughput of TCP/IP over ATM decreases rapidly when the Burst Tolerance is decreased, regardless the MTU size. The throughput never reaches the SCR of 3 Mb as expected. The number of dropped cells increase rapidly as the Burst Tolerance is decreased (see table C.2). Apparently, the TCP mechanisms and operations cause extra delay so the throughput is decreased.

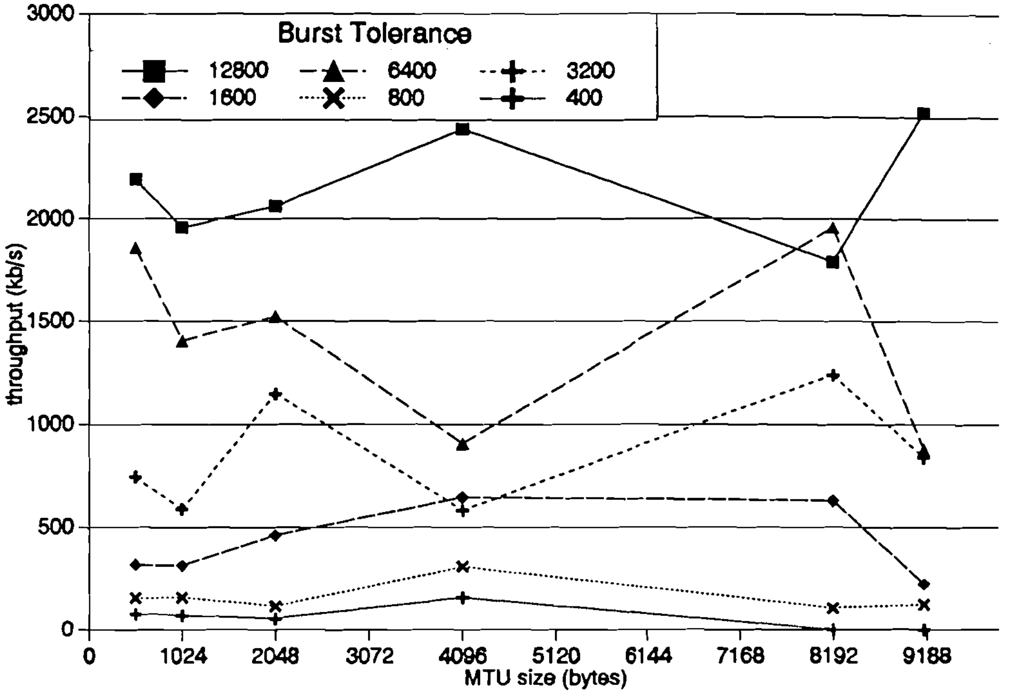


Figure 38 The throughput versus the MTU size for several BT values.

In figure 38 the throughput fluctuates as function of the MTU size. The throughput heavily depends on whether an equilibrium-state is reached quickly or not. When after a short time, the TCP protocol at the transmitter has reached a steady state, every second the same amount of cells is transmitted and the throughput is relatively high. However, the opposite situation occurs when the transmitter sends a large amount of cells, then a large percentage is discarded and as a result, the next few seconds no cells are transmitted. After some time a segment is transmitted successfully, which is followed by a large burst of cells. A steady state is never reached and the throughput is relatively low. Which situation is reached, (equilibrium or not) probably depends on the set MTU size and the BT. However, the exact reason is still unknown and for further study.

7.6.2 The influence of the peak rate on the throughput for extra traffic parameters

Again the same experiment configuration is used, including the PFB. At the PFB the peak rate is set to 10 Mb, the SCR is 3 Mb and the BT is 3200 cells. Also the same amount of User Data is transferred from EMMA2 to EMMA1 and at the EMMA workstations the MTU size is set to 8192 bytes, while the peak rate is varied between 3 and 10 Mb. The influence of the difference between the peak rate of the source and the SCR at the PFB on the throughput is examined. The results are depicted in table 12 and figure 39.

***Table 12** The throughput of TCP/IP over ATM versus peak rate for extra traffic parameters.*

peak rate (Mb)	*# cells*	*# datagrams*	*# drops*	*time (s)*	*throughput (kb/s)*
3.0	*116827*	*10624*	*0*	*16.5*	*2423.05*
4.0	*117203*	*10678*	*1803*	*30.5*	*1283.84*
5.0	*117018*	*10653*	*3270*	*32.0*	*1221.90*
6.0	*116832*	*10645*	*2773*	*55.7*	*715.38*
7.0	*117011*	*10648*	*2592*	*43.0*	*912.20*
8.0	*116951*	*10661*	*2314*	*70.7*	*554.09*
9.0	*116892*	*10659*	*2468*	*64.5*	*612.98*
10.0	*117039*	*10664*	*2803*	*56.0*	*697.55*

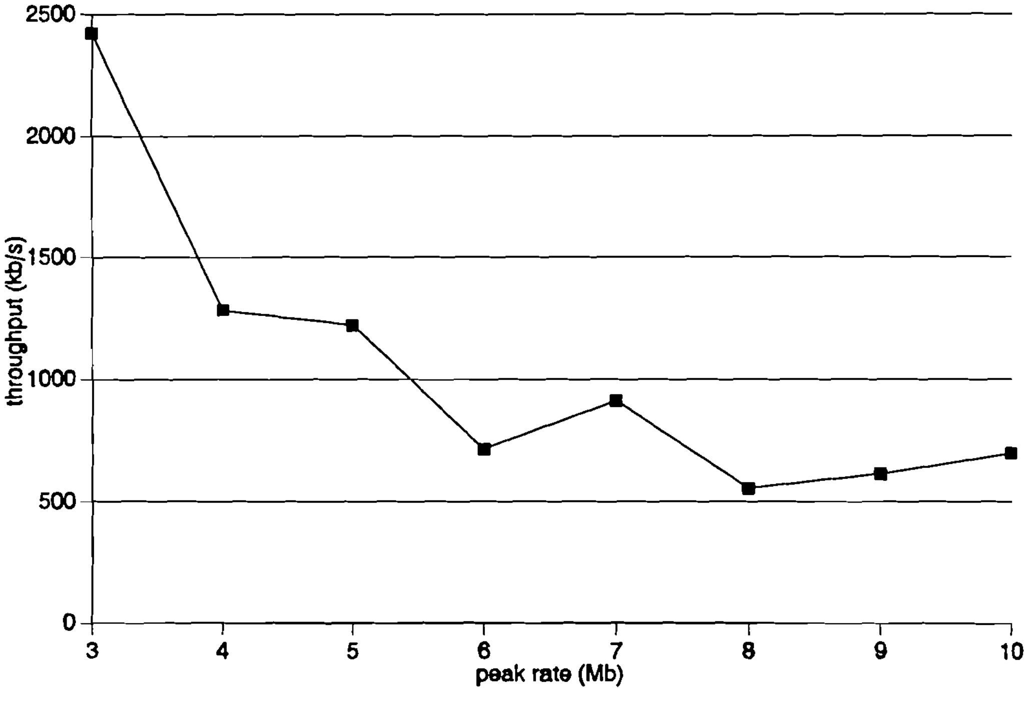


***Figure 39** Peak rate influences on the throughput for VBR traffic parameters.*

The highest throughput is achieved in case the peak rate at the EMMA workstations is equal to the SCR. In that case no cells are discarded by the PFB, since the BLM is never reached. When the peak rate of the workstations is increased, the throughput decreases. For higher peak rates, larger bursts of cells can be send, but the PFB will discard some cells and the TCP actions cause a decrease in throughput.

Regarding the experiments in paragraph 7.6.1 and 7.6.2, it can be concluded that policing with extra traffic parameters is not sensible for bulk data transport. Policing with the peak rate only gives a better performance than taking the SCR and the BT into account.

For smaller amounts of User Data, i.e. smaller than the BT, this way of policing can be beneficial. Suppose a number of small files, pictures for example, has to be transferred with some interval period. When the small files fit into the BT, a file can be sent at peak rate every time, while a lower SCR has been set.

7.7 Experiment results and conclusions

In this chapter a large number of ATM network experiments is described and analyzed. The experiment results and the conclusions are listed in this paragraph.

Performance of UDP/IP and TCP/IP over ATM

- The segmentation of User Data into UDP datagrams or TCP segments and the translation into ATM cells via AAL5 is examined in detail and complies with the Recommendations and RFCs.

- The throughput of UDP/IP traffic over ATM is conform the peak rate settings at the workstation. The maximal throughput is about 80 Mb/s, even for higher peak rate settings.

- At a small time scale, not during the start up or finish period, the maximal throughput of UDP/IP traffic over ATM with the SunSPARC20 EMMA2 workstation as source is significantly higher than with the SunSPARC10 EMMA1 workstation as source (98.6 Mb/s versus 70.8 Mb/s).

- In case of data transport via UDP/IP or TCP/IP over ATM, using smaller datagrams (smaller MTU size) results in a more rapid decrease of the throughput than is expected on base of the extra amount of overhead and ATM cells. This is caused by the extra processing time needed to generate the datagrams and ATM cells. A specification of this processing time is given in Annex A.

- For both UDP/IP and TCP/IP over ATM, the largest MTU size results in the highest throughput in case no cell loss occurs.

- The throughput of TCP/IP traffic over ATM is conform the peak rate settings at the workstation only for low values (less than 45 Mb/s). For higher peak rate settings the maximal throughput remains about 45 Mb/s. So the throughput of TCP/IP over ATM is significantly lower than the throughput of UDP/IP over ATM.

Influence of cell loss on the throughput of TCP/IP over ATM

- In case cell loss occurs, the throughput of TCP/IP over ATM decreases more quickly than is expected on base of the number of retransmitted cells. Due to TCP algorithms about 0.9 to 1 second extra delay is introduced when cell loss occurs.

- In case of a Cell Loss Ratio larger than 10^{-6}, the throughput of TCP/IP over ATM decreases significantly.

- For a Cell Loss Ratio of 10^{-5}, an MTU size of 8192 bytes gives a better performance than larger MTU sizes. The MTU size is nonessential in case the Cell Loss Ratio becomes 10^{-4} or more.

Influence of extra traffic parameters on the throughput of TCP/IP over ATM

- Including extra traffic parameters besides the peak rate, like SCR and BT, in the policing scenario is not sensible for bulk data transport.

- The highest throughput is achieved in case the peak rate at the EMMA workstations equals to the SCR. In that case no cells are discarded by the PFB, since the BLM is never reached. When the peak rate of the workstations is increased, the throughput decreases. For higher peak rates, larger bursts of cells can be send, but the PFB will discard some cells and the TCP actions cause a decrease in throughput.

- When extra traffic parameters are included in the policing scenario, the throughput of TCP/IP over ATM decreases quickly when the BT value is decreased, regardless of the MTU size. Only for large BT values (more than 12800 cells), the throughput can come close to the set SCR.

8 Multiplex experiments

8.1 Introduction

In the previous chapter, ATM network experiments are performed without multiplexing being involved. Next a number of multiplex experiments is described and analyzed. The overall goal of these multiplex experiments is to examine the usefulness of statistical multiplexing for TCP/IP and UDP/IP traffic. Furthermore the influence of multiplexing on the NP and QoS will be examined. When a number of sources is multiplexed, the overall and individual performance can be measured and will be compared to the performance of a single TCP/IP or UDP/IP traffic source in chapter 6.

First two UDP/IP data streams containing small bursts are multiplexed. The GDC switch used for these multiplex experiments was assumed to contain an output buffer of 1171 cells deep. However, in Annex D this buffer length is measured, which resulted in a buffer of only 31 cells deep. For multiplex experiments such buffer is too small, since cell loss will often occur.

In the near future the present C series cards in the GDC switch will be replaced by H series cards, containing a dual output buffer of 63 and 1171 cells. Another solution is to include the ATC in the experiment configuration. Once the buffer problem is solved, a number of useful multiplex experiments can be performed. In the last paragraph these future experiments are described in theory only and it is recommended to perform them later.

8.2 Multiplexing UDP messages without peak rate limitation

The first multiplex experiment that is performed, includes the multiplexing of two sources into a link. The experiment configuration is given in figure 40. Two sources, EMMA2 and Silicon2, send UDP messages that are transmitted using ATM. The traffic is multiplexed into a 100 Mb/s link by the GDC switch and received by EMMA1.

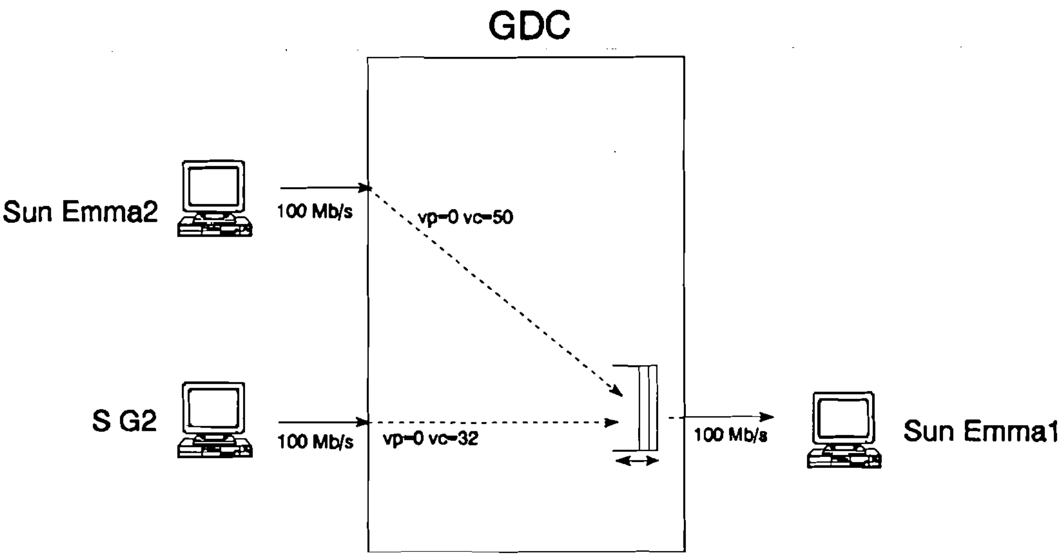


Figure 40 *Multiplex experiment configuration.*

The sources contain no peak rate limitation functionalities, so two data streams with a peak rate of 100 Mb/s are multiplexed to a link of also 100 Mb/s. In case the total input should temporarily exceed this output rate, the buffer in the GDC switch will temporarily store the extra cells. However, cells that can not be stored in this buffer will be lost. The goal of this experiment is to examine the influence of the burst sizes and time intervals between the bursts on the amount of cell loss.

At EMMA2 bursts are generated using `udpt`, so the burst size and interval between the bursts can be varied. For Silicon2 a similar program, called `spray`, is used that has the following parameters:

- The burst size *'l'* (in bytes)
- The interval between the bursts *'d'* (in micro-seconds)
- The number of bursts *'c'* to be transmitted (an integer)

8.2.1 Multiplexing of small bursts with large time intervals

First small bursts with large time intervals are multiplexed into a link. Then the burst length is increased, and also the interval times are increased, i.e. the average cell rate is not changed. The probability of cell loss due to buffer overflow is expected to be small. At Silicon2 the command `spray -c 50000 -l 8192 -d 1000000 emma1` is given, which results in a burst of 171 cells once per second.

At EMMA2 a burst of 23 cells once per second are generated using `udpt -n 50000 -t 1000000 -s 1024 emma1`. EMMA1 receives 194 cells per second and no cells are dropped.

Now the burstlength and the interval time at EMMA2 are doubled, so the ACR at EMMA2 remains about 23 cells/s, while the command at the Silicon2 is not changed. The results are listed in the table 13.

■ ***Table 13** Multiplexing of small bursts with large intervals coming from two sources.*

Source EMMA2			*Source Silicon2*	*Destination EMMA1*	
n (bytes)	*t (s)*	*cell pattern / s*	*cells / s*	*Cells / s*	*Drops*
1024	*1*	*23,23,23,23,..*	*171,171..*	*194,194,194,194..*	*0*
2048	*2*	*44,0,44,0,44,0,..*	*171,171..*	*215,171,215,171..*	*0*
4096	*4*	*87,0,0,0,87,0,0,0,87..*	*171,171..*	*258,171,171,171,258..*	*0*
8192	*8*	*172,0,0,..,0,172,0..*	*171,171..*	*343,171,..,171,343,171..*	*0*

Multiplexing two sources that generate small bursts with large time intervals results in no cell loss. This result corresponds with the expectations, since the probability that two bursts arrive at the GDC switch exactly at the same time is very small. Besides, the cells in a burst are not transferred back-to-back, since an intercell gap exits (see Annex B) and this is seen during measurements at ATM cell level ([Dijk95]). In the empty spots between the cells, temporarily buffered cells can be transported.
The interval times are more than large enough to empty the buffer.
However, since the buffer size is only 31 cells, cell loss can occur when two larger bursts are multiplexed and arrive exactly at the same time. During this measurement period (about 5 minutes) no cell loss occurred.

8.2.2 Multiplexing of small bursts with small time intervals

A similar multiplexing experiment is performed, but the Average Cell Rate (ACR) of the EMMA2 and Silicon2 workstation are increased compared to 8.2, 8.2.1. The burst sizes remain approximately the same, while the time intervals between the bursts are decreased. Now the probability that two bursts arrive at the switch at the same time or at least have some overlap has increased.

Using the command `spray -c 10000 -1 5000 -d 100000 emma1`, Silicon2 generates bursts of 5000 bytes (105 cells) with an interval of 0.1 second, so the ACR is 1050 cells/s. The results are listed in table 14.

■ ***Table 14*** *Multiplexing of bursts with a larger ACR coming from two sources.*

Source EMMA2				*Source Silicon2*	*Destination EMMA1*	
n (bytes)	*cells*	*t (s)*	*cells/s*	*cells/s*	*Cells/s*	*Drops*
1024	*23*	*0.125*	*8x23*	*10x105*	*1234*	*0*
2048	*44*	*0.25*	*4x44*	*10x105*	*1226*	*0*
4096	*87*	*0.5*	*2x87*	*10x105*	*1224*	*255*
8192	*172*	*1*	*1x172*	*10x105*	*1222*	*218*

When bursts of more than 44 cells are multiplexed with bursts of 105 cells, one drop of 255 respectively 218 cells occurs. For smaller bursts no cell loss occurs, even though the cells received per second is equal. Thus, in order to avoid cell loss during multiplexing, one should send small bursts more frequently; this is called smoothing, spacing or peak rate limitation. The cell loss occurs for small bursts already since the buffer in the GDC switch is only 31 cells deep. In case a larger buffer than 31 cells is implemented in the switch, larger bursts with smaller interval times can be multiplexed. It is useless to execute other multiples experiments with this configuration. Either a larger buffer should be implemented or another solution for the buffer problem has to be found. When these problems are solved, the multiplex experiments should be continued and therefore a number of interesting future experiments is described in the next paragraph.

8.3 Future multiplex experiments

Due to the presently small buffer in the GDC switch, the multiplex experiments could not be performed. In case the H-series cards are implemented in the GDC switch a larger buffer can be used for multiplex experiments. Another solution is to connect an Adaptive Traffic Controller (ATC), [Wal95b], to a switch and to use this shaper as multiplexing buffer (see figure 41). Since the interfaces at the ATC did not match with the interfaces at the GDC or PKI switch, these experiments could not be performed. However, in future, the technical problems might be solved and the following multiplex experiments are recommended to perform.

The output capacity of the ATC and the buffer size in the ATC can be controlled. The GDC and PKI switch contain a small output buffer, so the peak rate of the sources should be limited to avoid cell loss in the switches. Two SunSPARC workstations and two Silicon Graphics workstations are connected to the GDC switch, while four PC's can be used as data sources or receiver at the PKI switch.

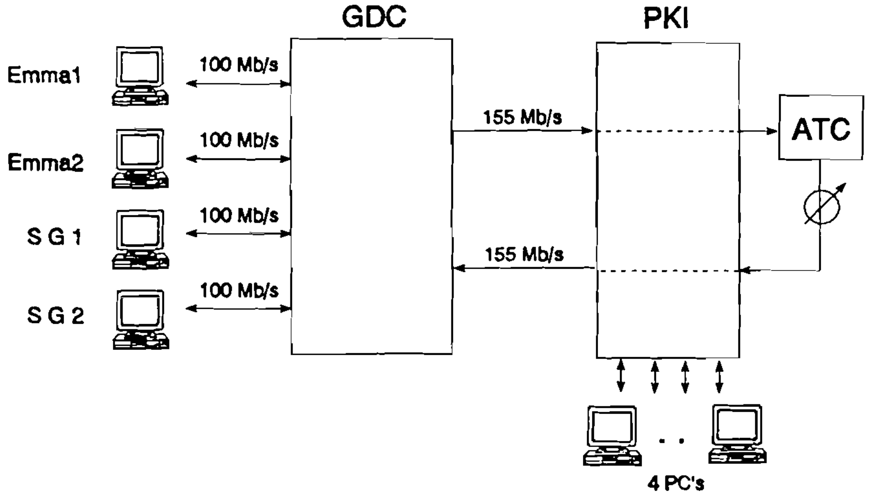


Figure 41 Multiplex experiment configuration using the ATC.

In order to employ the Silicon Graphics workstations for multiplex experiments, new ATM adapter cards including peak rate limitation have to be implemented. Otherwise cell loss will already occur in the GDC switch. The PC's can only generate traffic with a peak rate of about 4 Mb/s.

8.3.1 Multiplexing UDP messages

Using the new multiplex experiment configuration in figure 41, the previous multiplexing of UDP messages is be repeated. Only this time more sources can be used, the buffer size and the output capacity can be varied. First two sources (EMMA1 and EMMA2) with a peak rate of 40 Mb/s are multiplexed, while the buffer size in the ATC is maximal (i.e. 30,000 cells) and the output capacity is the sum of the peak rates (80 Mb/s). Then the output capacity is decreased and the MTU size is varied. The overall and individual throughput and the amount of cell loss is measured. This experiment is repeated for smaller buffers, e.g. 1171 cells as in the H-series cards of the GDC switch and about 400 cells, so that maximally 2 datagrams can be buffered.

8.3.2 Multiplexing TCP messages

The experiment configuration remains the same (see figure 41), only this time TCP messages are multiplexed. Now cell loss due to multiplexing initiates retransmissions of complete segments (large number of cells). Again first two sources are multiplexed with a peak rate of 40 Mb/s. In paragraph 7.5.5 is concluded that for peak rates below 45 Mb/s the throughput is still conform the set peak rate.

For higher peak rates the throughput is limited by the workstations. The output capacity of the ATC is set to 80 Mb/s (peak rate allocation), while the MTU size is varied. Also the buffer size is varied between the maximal value (30,000 cells), the buffer size of the GDC switch (1171 cells) and a small buffer (about 400 cells). The transfer delay and throughput are measured. It is expected that the throughput is identical as in paragraph 7.5.5.

When the output capacity of the ATC is decreased to 40 Mb/s, while at the sources nothing has changed, the throughput is expected to decrease. It is expected that the throughput will decrease more than 50% when the output capacity is halved. Cell loss will occasionally occur and initiate retransmissions of complete segments, window sizes will decrease and TCP algorithms are performed. These actions cause extra delay, so the throughput will decrease extra. The influence of the output capacity value of the ATC on the throughput is examined.

Next, the number of sources is extended to 6 sources, including the PC's in the experiment. First the capacity is divided homogeneously, so the peak rate at the workstations is set to the PC's peak rate (4 Mb/s). The output capacity of the ATC is set to 24 Mb/s and slowly decreased, while the MTU size is varied again. This is also repeated for smaller buffers.

Secondly the capacity is divided inhomogeneously over the sources; the EMMA workstations produces data with a peak rate of 20 Mb/s, while the peak rate at the PC's is only 4 Mb/s. The output capacity at the ATC is 56 Mb/s and slowly decreased, while the MTU size is varied again. The throughput is measured, and the difference between a homogeneous and inhomogeneous distribution of the output capacity of the ATC is analyzed.

To examine the usefulness of real statistical multiplexing, the number of traffic sources in these experiments is still insufficient. Maybe other traffic sources that are connected to the PKI switch, like HDTV monitors, video monitors, the FDDI LAN and the N-ISDN gateway, can be included in the experiment configuration.

9 Conclusions and recommendations

In this chapter the conclusions are presented together with a number of recommendations in the form of a list of several items that can be used for further study.

9.1 Conclusions

In this report the performance of TCP/IP over an operational ATM network is examined. Experience has been gained with the available ATM equipment and ATM traffic control utilities. The segmentation of User Data into UDP datagrams or TCP segments and the translation into ATM cells via AAL5 is examined in detail and are conform Recommendation I.363 and RFC 791.

Generally, it can be concluded that, for sufficiently low peak rate settings at the data source, a large MTU size, and a low Cell Loss Ratio, the throughput of UDP/IP and TCP/IP over ATM complies with the expectations based on theoretical considerations.

However, for high peak rate settings at the data source, the throughput becomes lower than expected. This is not caused by the principle of TCP/IP and UDP/IP over ATM, but by the limited processing power of the workstations in stead. The throughput of UDP/IP over ATM is significantly higher than the throughput of TCP/IP over ATM for high peak rate settings, since less processing power is required for the UDP protocol.

When the MTU size is decreased significantly, the throughput of both UDP/IP and TCP/IP over ATM diminishes more than is expected based on the extra amount of overhead and the total number of transmitted cells.

Furthermore, the throughput of TCP/IP over ATM decreases seriously, in case the Cell Loss Ratio exceeds 10^{-6}[*]. The impact of smaller MTU sizes and cell loss on the throughput is caused by extra processing delay, consisting of the interdatagram gap, intercell gap and an extra delay component due to cell loss. Values for these delay components are estimated and can be employed in order to offer reliable TCP/IP connections over ATM in the near future.

Including extra traffic parameters, like Sustainable Cell Rate and Burst Tolerance, besides the peak rate in the policing scenario is not sensible for bulk data transport. However, for the transport of small amounts of data (like pictures) that fit into the Burst Tolerance, this way of policing may be favourable.

No thorough conclusions can be drawn about the usefulness of statistical multiplexing for UDP or TCP data streams, since the proper experiments could not be performed. The available output buffer of 31 cells in the GDC switch is too small to perform statistical multiplexing.

9.2 Items for further study

General open issues

- In this report the performance of data transfer applications using TCP/IP over ATM is examined. However, the performance might be improved when a new data transfer protocol especially for ATM is designed, or the TCP/IP suite is adapted to ATM.

  One of the causes of the rapid performance decrease of TCP/IP over ATM when cell loss occurs, is the slow start algorithm of TCP. This algorithm is implemented in TCP to initiate data flow and avoid congestion in the network (see paragraph 5.5.2). However, since ATM is connection oriented and sufficient bandwidth should be allocated, the slow start algorithm might be redundant. This is for further study. It should be noted that the CLR in the performed experiments was constant. In practise, when congestion occurs, the CLR will decrease after TCP has reduced the traffic load.

- In this report the throughput is considered to be the amount of User Data arrived at the receiver divided by the transfer time, i.e. an QoS parameter. For the network operator, however, it is also interesting to examine the total amount of User Data transferred through the network.

[*] *This is no random cell loss, but one in every million cells is intentionally discarded.*

Due to retransmissions this amount of User Data might differ from the received amount of User Data and the network provider's point of view on the throughput might differ from the user's experience. The network provider's point of view on the throughput is a parameter of the performance of the network and is important for an efficient exploitation of an ATM network. In this case, tariffing is also an interesting subject that should be further examined. The customer will only pay for the received amount of User Data, while the network provider has transferred a possibly larger amount of User data.

Open issues based on the performed experiments

- In paragraph 7.5.6, the influence of Cell Loss on the throughput of TCP/IP over ATM is considered. The overall extra delay due to Cell Loss is estimated to be around 0.9 seconds. Using the HP tester a trace at ATM cell level can be monitored and the influence of each TCP algorithm separately can be distinguished. This way the influence of each TCP algorithm on the throughput can be deepened out.

- The cell loss experiments of paragraph 7.5.6 should be repeated for larger RTTs, e.g. a connection can be set up between Leidschendam and Sweden. It is expected that the influence of cell loss on the throughput of TCP/IP over ATM becomes larger, since it takes more time to receive ACKs and therefore the slow start, fast retransmit and fast recovery algorithms are slowed down.

- The cell loss in the experiments of paragraph 7.5.6 occurred not randomly, but was initiated by the PFB. In case the CLR is set to be 10^{-3}, each time one cell is discarded after 999 cells have passed. In practise, however, cell loss is mainly caused by buffer overflow and will therefore occur in groups. Since AAL5 discards all cells belonging to a datagram when cells are lost, the loss of a group of cells belonging to one datagram will have the same performance degradation as the loss of one cell, while the CLR might be much higher. For example, a situation with a CLR of 10^{-6} where one in a million cells is lost and a situation with a CLR of 10^{-4} where a group of 100 cells belonging to the same datagram in a million cells is lost, have the same performance degradation. In future cell loss experiments this should be kept in mind.

- The MTU size at IP layer was limited to 9188 bytes during the performed experiments, but in future this maximal MTU size might be increased in order to improve the throughput of TCP/IP over ATM. The advantage of larger MTU sizes is that less datagrams are needed to transfer an amount of User Data, which results in less interdatagram gaps and therefore less processing time. The disadvantage is, however, that the influence of cell loss on the throughput will increase.

The efficiency will not significantly be improved in case of larger MTU sizes, since an MTU of 9188 bytes already results in an efficiency of 89.9%, while ATM has a maximal efficiency of 90.6%.

- In case extra traffic parameters are considered, like in paragraph 7.6, the throughput is relatively high when an equilibrium state is reached and low when this steady state is never reached. Which state is reached, equilibrium or not, might depend on the set MTU size and the BT. However, the exact reason is still unknown and for further study.

- In paragraph 8.3 a number of future multiplex experiments is described. When the ATC or the H series cards in the GDC switch are available, these proposed experiments can be performed and analyzed.

Recommendations regarding the available equipment at KPN Research during the ATM network experiments

- The C series cards in the GDC switch only contain a 31 cells deep output buffer. For a number of experiments and surely for multiplex experiments, this output buffer is too small and therefore it is desirable to implement the H series cards soon. The H series cards contain two independent output buffers of 63 and 1171 cells deep.

- The FORE 100 ATM cards at the Silicon Graphics workstations cause cell drops when large bursts are received. Therefore these ATM cards should be replaced by FORE 200 ATM cards.

- For data transmission using UDP and low peak rate settings at the EMMA workstation used as source, User Data is lost. The limited buffer in the FORE 200 ATM card (64 KB) is suspected to be the cause. The workstation will generate the User Data as fast as possible, but the ATM FORE 200 card has to buffer data in order to comply with the set peak rate. It is desirable to implement a control signal that notifies the source that the buffer is full and the data generation has to be stopped temporarily. Probably this should be implemented in the C program `udpt` used to generate the User Data.

- The Silicon Graphics workstations contain no C compiler and therefore can not run the C programs `udpt`, `udpr` and `ttcp`, like the SunSPARC stations (EMMAs). This might become a problem during multiplex experiments.

- An improved user interface of the Police Function Board in order to simplify the use of the PFB is recommendable. For choosing a policing scenario and setting a CLR value, the help of an experienced PFB user is needed.

- The connection management and the management of the switches should become less complex. The GDC switch contains a GDC Manager that is used to establish a connection. However, a large number of steps have to be taken before a connection is set up in the switch or before a connection is changed. This can be facilitated using a management application (like for example NMS3000).

- The communication between all projects using the same experiment equipment should be improved. When a number of experiments is performed at the same time, the experiments might affect each other and thus decrease the accuracy of the measurements. Therefore the required equipment should be reserved (only for the actual experiment period) and the other project-members should be informed. Since experiments are executed often via a remote login, a warning should be displayed at the terminal of the workstations.

- Sufficient test equipment should become available, including a HP tester containing Cell Protocol Processor (CPP) modules in order to make traces at ATM cell level.

- In order to examine the usefulness of statistical multiplexing, a large number of data sources is necessary. With the available number of sources in chapter 8 probably only an indication of the usefulness can be given.

References

[Ana91] M. Anagnostou, M. Theologou, K. Vlakos, D. Tournis, and E. Protonotarius, *Quality of service requirements in ATM-based B-ISDNs*
Computer Communications vol 14 no 4, May 1991, pag. 197-204

[App91] J. Appleton, *Performance related issues concerning the contract between network and customer in ATM networks*
BT Technology Journal Vol 9 No 4 October 1991

[Arm95] G.J. Armitage, and K.M. Adams, *How efficient is IP over ATM anyway?*
IEEE Network, January/February 1995, pag. 18-26

[ATMF94] The ATM Forum Technical Committee, *User-Network Interface (UNI) Specification Version 3.1*
September, 1994

[Black94] U. Black, *TCP/IP and Related Protocols*
McGraw-Hill, Inc., 1994
ISBN 0-07-005560-2

[Bri94] A. Brinkmann, J. van Dijk, C. Lavrijsen, R. Lehnert, T. Müller, M. Olesen, *Initial TRIBUNE test-bed performance evaluation*
Deliverable No. R/R/4.1/08, December 1994-Issue 1

[Chao94] H.J. Chao, D. Ghosal, D. Saha, and S.K. Tripathi, *IP on ATM Local Area Networks*
IEEE Communications Magazine, August 1994, pag. 52-59

[Dijk95] J. van Dijk, M. Emons, C. Lavrijsen, R. van der Mei, P. Venemans, K. van der Wal, *ATM Pilot performance experiments*
R&D-RA-95-XXXX

[Dirk91] M.J.G. Dirksen, *Traffic control and resource management in ATM networks*
NT-ER-91-1439

[Feit93] S. Feit, *TCP/IP: architecture, protocols, and implementation*
McGraw-Hill, Inc., 1993
ISBN 0-07-020346-6

[GDC94] General DataComm, *ATM and Adaption Switch Family APEX*
APEXprd2.doc, Amsterdam, November 14, 1994

[Gil91] H. Gilbert, O. Aboul-Magd and V. Phung, *Developing a cohesive traffic management strategy for ATM networks*
IEEE Communications Magazine, October 1991, pag. 36-45

[Händ94] R. Händel, et al., *An introduction to ATM-based networks*
Addison-Wesley, 1991
ISBN 0-201-54444-X

[I.356] ITU-T Recommendation I.356: *B-ISDN ATM layer cell transfer performance*
Helsinki, March 1993

[I.361] ITU-T Recommendation I.361: *B-ISDN ATM layer Specification*
Geneva 1992

[I.362] ITU-T Recommendation I.362: *B-ISDN ATM Adaptation Layer (AAL) Functional Description*
Geneva 1992

[I.363] ITU-T Recommendation I.363: *B-ISDN ATM Adaptation Layer (AAL) specification*
Helsinki, March 1993

[I.371] ITU-T Recommendation I.371: *Traffic control and congestion control in B-ISDN*
Helsinki, March 1993

[Jac88] V. Jacobson and M.J. Karels, *Congestion Avoidance and Control*
Computer Communication Review, vol. 18, no. 4,
August 1988, pag. 314-329

[Jeff94] R. Jeffries, *ATM LAN Emulation: The inside story*
Data Communications, September 1994, pag. 95-100

[Jung93] Jae-il Jung and D. Seret, *Translation of QoS parameters into ATM performance parameters in B-ISDN*
Infocom '93, pag. 748-755

[LESG94] ATM Forum Technical Committee, *(Keep it simple) LAN Emulation Tutorial*
LAN Emulation Sub-working Group
ATM_Forum 94-0225, March 1994

[New94] P. Newman, *ATM Local Area Networks*
IEEE Communications Magazine, March 1994, pag. 86-98

[Pry91] M. de Prycker, *Asynchronous Transfer Mode*
Ellis Horwood Limited, 1991
ISBN 0-13-053513-3

[RFC791] University of Southern California, *Internet Protocol*
Information Science Institute, September 1981

[RFC1122] R. Braden, *Requirements for Internet Hosts -- Communication Layers*
Internet Engineering Task Force, October 1989

[RFC1577] M. Laubach, *Classical IP and ARP over ATM*
Hewlett-Packard Laboratories, January 1994

[Sim93] N. Simoni, and S. Znaty, *QoS: from definition to management*
High Performance Networking IV (C-14), Elsevier Science Publishers B.V., IFIP 1993

[Smits93] T.A. Smits, *The poor gain from statistical multiplexing in the homogeneous and the heterogeneous case*
IBCN&S, Copenhagen, April 19-23 1993, article 13.1

[Stev94] W.R. Stevens, *TCP/IP illustrated, volume 1*
Addison-Wesley publishing company, November 1994
ISBN 0-201-63346-9

[Voo94] D.A. Voogt, P.F.C. Blankers, *Ethernet and ATM: How to live in harmony ?*
R&D-RA-94-848

[Vries94] Dr.ir. R. de Vries, dr. A. van Blokland, *ATM: A status Report - issue 5*
R&D-RA-94-794, October 1994

[Wal94] Ir. J.C. van der Wal, *Zin en onzin over spraak via ATM*
RA-94-564, June 1994

[Wal95a] K. van der Wal, *Traffic shaping of FORE SBA-200 AAL5 cards in EMMA workstations*
Internal document, March 28th 1995

[Wal95b] K. van der Wal, *Some quick notes on setting the ATC parameters*
Internal document, June 6th 1995

Annex A Calculation of workstation's processing time

In figure 25 the throughput of UDP/IP over ATM decreases rapidly when the MTU size is decreased. It is expected that the processing delay of the workstations EMMA1 and EMMA2 is the cause of this decrease. The processing delay consist of the interdatagram gap, which is the time needed by the workstation to generate a UDP datagram and the intercell gap, which is the time needed to translate a UDP datagram into ATM cells, see figure 42.
Using the measurements of paragraph 7.4.4, an estimation of the interdatagram gap and the intercell gap is made for both workstations.

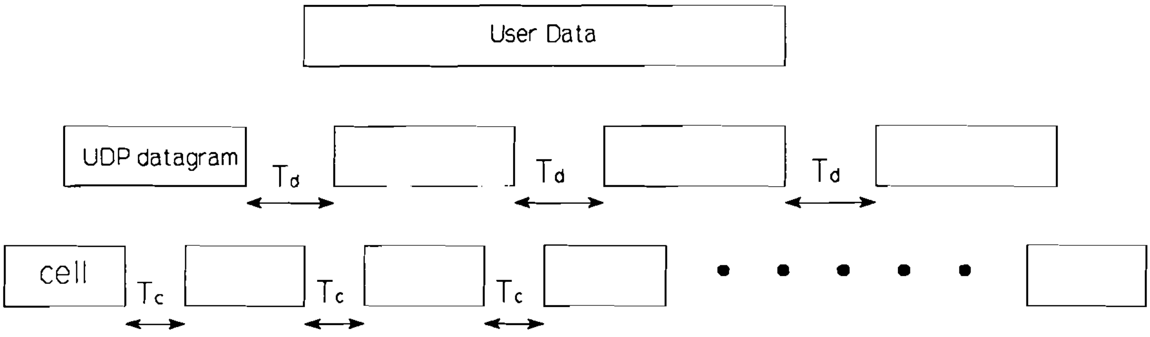


Figure A.1 Interdatagram gap (T_d) and Intercell gap (T_c).

In paragraph 7.4.6 the maximal throughput measured for respectively the EMMA2 and EMMA1 workstation appears to be about 98.6 Mb/s and 70.8 Mb/s in stead of the theoretical value of 100 Mb/s. Using these measured bit rates and the number of cells, the intercell gap can be calculated for EMMA1 and EMMA2, see table A.1.

Table A.1 *Calculation of the intercell gap.*

		EMMA1 (70.8 Mb/s)			*EMMA2 (98.6 Mb/s)*		
# cells	*T_{theory} (s)*	*T_{E1} (s)*	*difference*	*T_c (µs)*	*T_{E2}*	*difference*	*T_c (ns)*
5,160,000	*21.88*	*30.90*	*9.02*	*1.70*	*22.20*	*0.32*	*62*
5,190,000	*22.01*	*31.08*	*9.07*	*1.70*	*22.33*	*0.32*	*62*
5,220,000	*22.13*	*31.26*	*9.13*	*1.70*	*22.46*	*0.33*	*63*
5,250,000	*22.26*	*31.44*	*9.18*	*1.70*	*22.59*	*0.33*	*63*
5,430,000	*23.02*	*32.52*	*9.50*	*1.70*	*23.36*	*0.34*	*63*
5,550,000	*23.53*	*33.24*	*9.71*	*1.70*	*23.88*	*0.35*	*63*

T_{theory} is the minimal theoretical transfer delay, based on a bit rate of 100 Mb/s. In this case no intercell gap occurs, so all cells lay back-to-back. In practise, however, for EMMA1 and EMMA2 another bit rate is measured and this results in a different transfer delay (T_{E1} and T_{E2}), that can be calculated based on the measured number of cells; expected transfer delay $= \frac{\text{number of cells x 53 x 8}}{\text{max bit rate}}$

The difference between T_{theory} and T_{E1} or T_{E2} is the cell processing delay, so

$$\text{intercell gap} = \frac{T_{theory} - T_{E\,i}}{\text{number of cells}} \qquad ,i = 1,2$$

According to the values in table A.1, the intercell gap is respectively 1.70 µs for workstation EMMA1 and 63 ns for workstation EMMA2.

Performing a similar calculation, the interdatagram gap for EMMA1 and EMMA2 can be determined; interdatagram gap $= \frac{T_{E\,i} - T_{measure}}{\text{number of datagrams}} \qquad ,i = 1,2$

The difference between this expected transfer delay, based on the measured bit rates (T_{E1} and T_{E2}) and the measured transfer delay is the processing time needed to generate the datagrams. The average time needed to generate one datagram is calculated to be about 90 µs for EMMA2 and about 110 µs for EMMA1, see table A.2.

■ ***Table A.2*** *Calculation of the interdatagram gap.*

EMMA2 (transmit rate 98.6 Mb/s)				*EMMA1 (transmit rate 70.8 Mb/s)*			
$T_{measure}$ *(s)*	T_{E2} *(s)*	*processing delay (s)*	T_d *(µs)*	$T_{measure}$ *(s)*	T_{E1} *(s)*	*processing delay (s)*	T_d *(µs)*
24	*22.20*	*1.80*	*60.0*	*33*	*30.90*	*2.10*	*70.0*
28	*22.33*	*5.67*	*94.5*	*38*	*31.08*	*6.92*	*115.3*
30	*22.46*	*7.54*	*83.8*	*41.5*	*31.26*	*10.24*	*113.8*
37	*22.59*	*14.41*	*96.1*	*50*	*31.44*	*18.56*	*123.7*
48	*23.36*	*24.64*	*91.3*	*64*	*32.52*	*31.48*	*116.6*
70	*23.88*	*46.12*	*90.4*	*87*	*33.24*	*53.76*	*105.4*
			average 86.0 µs				*average 107.5 µs*

Finally, it is verified that the measurement results comply with estimated intercell gap and interdatagram gap, see table A.3. The transfer delay now is based on the theoretical 100 Mb/s bit rate.

■ ***Table A.3*** *Calculated versus measured transfer delays for EMMA2.*

MTU	*transfer delay (s)*	*interdatagram gap (s)*	*intercell gap (s)*	*Calculated delay (s)*	*Measured delay (s)*
9188	*21.88*	*2.70*	*0.325*	*24.9*	*24*
8192	*22.01*	*5.40*	*0.327*	*27.7*	*28*
4096	*22.13*	*8.10*	*0.329*	*30.5*	*30*
2048	*22.26*	*13.50*	*0.331*	*36.1*	*37*
1024	*23.02*	*24.30*	*0.342*	*47.7*	*48*
512	*23.53*	*45.90*	*0.350*	*69.8*	*70*

Table A.3 shows that the calculated delay, which is the transfer delay at 100 Mb/s rate + interdatagram gap + intercell gap, complies to the measured delay for the EMMA2 workstation. A similar verification is performed for EMMA1 with a positive result.

So it can be concluded that the processing time of EMMA1 workstation causes an interdatagram gap of 110 µs and an intercell gap of 1.7 µs. Similarly, the processing time of EMMA2 workstation causes an interdatagram gap of 90 µs and an intercell gap of 63 ns.

Annex B Cell drops in atmstat

In the previous experiments `atmstat` occasionally displayed a number of cell drops, but it is unknown what cell drops exactly indicates.
The monitor utility `atmstat -5 fa0 1` displays statistics about AAL5 traffic per second at the FORE Systems' ATM device driver. The fields of `atmstat` are as follows:

- Output cells: Number of cells transmitted by the ATM device driver.
- Input cells: Number of cells received by the ATM device driver.
- Drops: Number of cells dropped.
- CS-PDUs: Number of PDUs to (input) or from (output) CS-sublayer.
- Pay-CRC: Number of cells (AAL4) or CS-PDUs (AAL5) received with bad payload CRC.
- Congestn: Number of AAL5 CS-PDUs dropped due to cells lost or gained as a result of network congestion.

The number of dropped cells can be interpreted in two ways and at the moment it is unknown what `drops` indicates exactly.
Suppose one burst of 10 cells is sent and 2 cells get lost. At the receiver side an error check is performed, cell loss is detected and the segment will be discarded. `Atmstat` can either display 10 dropped cells (the expected number of cells) or 8 cells (number of the received and discarded cells). In the latter, `drops` is not equal to the lost cells in total. However, also the former explanation has its disadvantages. In case the trailer of a segment is lost, the receiver will wait for the next trailer to arrive, so a segment of 19 cells is received. Then a CRC is performed and the data will be discarded. In this case, `atmstat` will display 10 dropped cells, while in fact 2 segments (20 cells) are dropped. Another question is whether received and discarded cells are counted as `input cells` or not. An answer to these questions will be found for both the EMMA and the Silicon Graphics workstations.

Experiment at the EMMA workstations
This has been checked performing a simple experiment (see figure B.1).
EMMA1 sends an UDP message of 172 cells once per second to EMMA2, using the command `udpt -n 10 -t 1000000 -s 8192 emma2`. The UDP traffic is routed through the PKI switch, because the Police Function Board (PFB) is connected to this switch. At the PFB, the splash and Bucket Limit (BLM) are set to 32 and 32x498=159362, which results in a CLR of 2 10^{-3}. Now 499 cells are passed and 1 cell is discarded.
The results of `atmstat` are displayed in table B.1.

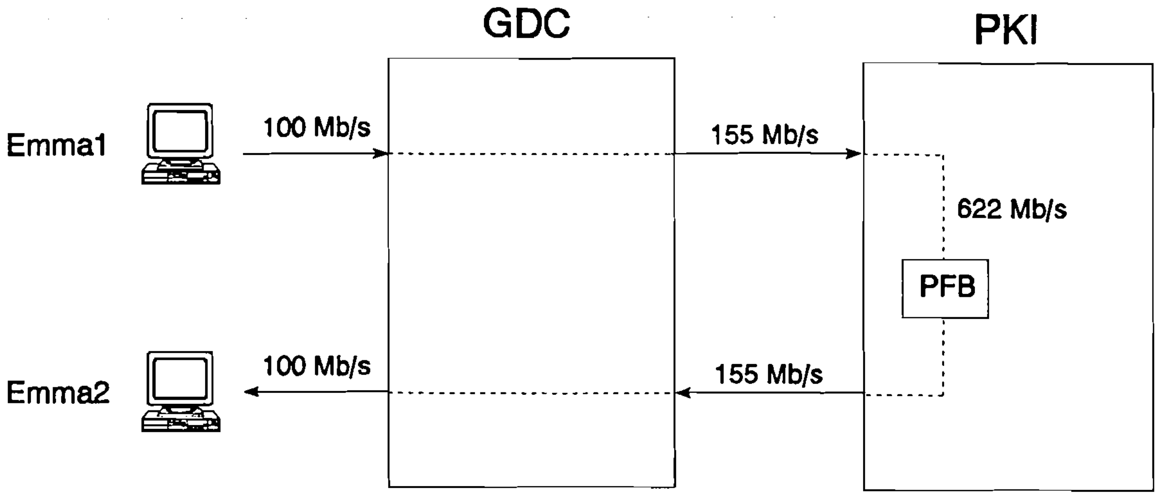


Figure B.1 *Drops in atmstat at the EMMA workstations.*

■ ***Table B.1*** *AAL5 statistics.*

Output at EMMA1		*Input at EMMA2*		*Errors at EMMA2*		
Cells	*CS-PDUs*	*Cells*	*CS-PDUs*	*Pay-CRC*	*Congestn*	*Drops*
172	*1*	*172*	*1*	*0*	*0*	*0*
172	*1*	*172*	*1*	*0*	*0*	*0*
172	*1*	*0*	*0*	*1*	*1*	*172*
172	*1*	*172*	*1*	*0*	*0*	*0*
172	*1*	*172*	*1*	*0*	*0*	*0*
172	*1*	*0*	*0*	*1*	*1*	*172*
172	*1*	*172*	*1*	*0*	*0*	*0*

Apparently `atmstat` displays the number of expected cells in a segment as drops in case one cell is discarded. The dropped cells are not registrated as received cells.

This experiment is repeated using other settings. The CLR now is 10^{-3} and at EMMA1 the command `udpt -n 10 -t 2000000 -s 1000 emma2` is given. This results in a cell stream of 22 cell every two seconds. As expected, after 91 seconds (990 cells are passed) 22 cells are dropped and also not registrated as received.

Experiment at the Silicon Graphics workstations

However, in case the same kind of experiment is performed at the Silicon Graphics workstations, the other possibility is seen.

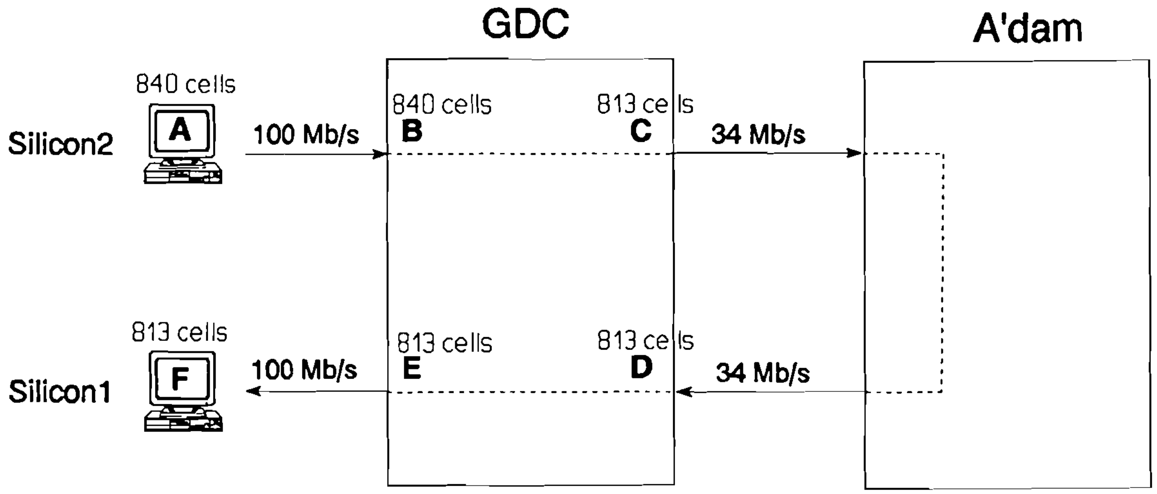


Figure B.2 *Drops in atmstat at Silicon Graphics workstations.*

Using the command `ping -s 2650 silicon1` at Silicon2, 56 ATM cells once per second are sent towards Silicon1 via the GDC switch and a loop to Amsterdam (see figure B.2).
Somewhere between the transmitter and the receiver, cells get lost (reason will be deepened out later) and `atmstat` displays the following:

■ ***Table B.2*** *AAL5 statistics.*

Output at silicon2		*Input at Silicon1*		*Errors at Silicon1*		
Cells	*CS-PDUs*	*Cells*	*CS-PDUs*	*Pay-CRC*	*Congestn*	*Drops*
56	*1*	*56*	*1*	*0*	*0*	*0*
56	*1*	*55*	*0*	*1*	*1*	*0*
56	*1*	*54*	*0*	*1*	*1*	*109*
56	*1*	*56*	*1*	*0*	*0*	*0*
56	*1*	*54*	*0*	*1*	*1*	*0*
56	*1*	*56*	*1*	*0*	*0*	*54*

In this case `atmstat` displays all cells received, but CS-PDUs shows whether a segment is complete or not. The not complete segments are discarded and that is displayed by `drops`, so lost cells are not included in `drops`.
So, it can be concluded that `drops` at `atmstat` has a different meaning at the EMMA workstations than at the Silicon Graphics workstations.
This should be kept in mind during experiments that involve both workstations.

Annex C Numerical experiment results

***■ Table C.1** Influence of Cell Loss on the throughput of TCP/IP over ATM.*

CLR	*MTU*	*# Cells*	*# PDUs*	*# Drops*	*# PDUs*	*Time (s)*	*Throughput (KB/s)*
0	9188	629656	3280	0	0	5	5859,38
0	8192	629319	3681	0	0	6	4882,81
0	4096	636119	7397	0	0	7	3348,21
0	2048	642587	14944	0	0	9	3255,21
0	1024	670849	30494	0	0	12	2441,41
0	512	701609	63783	0	0	21	1395,09
10^{-6}	9188	629664	3280	0	0	6	4882,81
10^{-6}	8192	629333	3681	171	1	7	4185,27
10^{-6}	4096	636112	7397	86	1	7	4185,27
10^{-6}	2048	642466	14941	0	0	9	3255,21
10^{-6}	1024	670800	30491	68	3	15	1953,12
10^{-6}	512	706515	64229	155	14	27	1085,07
10^{-5}	9188	629668	3280	1344	7	12	2441,41
10^{-5}	8192	629335	3681	1026	6	10	2929,69
10^{-5}	4096	636128	7397	516	6	13	2253,61
10^{-5}	2048	642464	14941	301	7	14	2092,63
10^{-5}	1024	670799	30491	132	6	18	1627,60
10^{-5}	512	699721	63611	280	26	33	887,784
10^{-4}	9188	629719	3280	12480	65	65	450,721
10^{-4}	8192	629438	3681	10944	64	64	457,764
10^{-4}	4096	636163	7398	5504	64	64	457,764
10^{-4}	2048	642473	14942	2752	64	64	457,764

10^{-4}	1024	670879	30495	1504	64	68	430,836
10^{-4}	512	699545	63595	958	87	74	395,904
10^{-3}	9188	21095	110	4893	26	26	37,560
10^{-3}	8192	21070	124	4362	26	27	36,169
10^{-3}	4096	21210	247	1978	23	23	42,459
10^{-3}	2048	21420	499	1031	24	22	44,389
10^{-3}	1024	22430	1020	550	25	22	44,389
10^{-3}	512	23344	2123	275	25	22	44,389
$2\ 10^{-3}$	9188	-		-	-	-	0,000
$2\ 10^{-3}$	8192	-		-	-	-	0,000
$2\ 10^{-3}$	4096	21295	248	4386	51	49	19,930
$2\ 10^{-3}$	2048	21506	501	2064	48	46	21,230
$2\ 10^{-3}$	1024	22579	1026	1172	53	45	21,701
$2\ 10^{-3}$	512	23463	2133	606	55	46	21,230

■ **Table C.2** Influence of the BT on the throughput of TCP/IP over ATM.

BT	MTU	cells	PDUs	drops	time (s)	throughput
12800	9188	105139	552	3645	15.5	2523.79
12800	8192	104888	617	3820	23.3	1790.30
12800	4096	106199	1242	3518	16	2441.41
12800	2048	107090	2501	2423	19	2061.63
12800	1024	112370	5120	1237	20	1958.02
12800	512	117512	10703	708	18.3	2193.73
6400	9188	104949	552	8858	44.5	877.92
6400	8192	104888	617	8196	20	1958.02
6400	4096	106045	1249	5906	44.5	877.92
6400	2048	107123	2497	4092	19	2061.63
6400	1024	111856	5102	2536	24.7	1628.2
6400	512	117651	10704	1088	20.5	1906.62
3200	9188	106495	569	13377	46.5	840.15
3200	8192	106769	628	12050	31.5	1240.39
3200	4096	106021	1237	8707	67.5	582.57
3200	2048	107682	2510	5152	34	1148.9
3200	1024	111825	5095	3964	68	587.15
3200	512	117120	10670	2683	52.5	744.12
1600	9188	104947	551	29297	176	221.95
1600	8192	109849	648	16265	62	630.2
1600	4096	108859	1270	11978	60.5	645.71
1600	2048	107082	2497	8642	85	459.62
1600	1024	111953	5097	5778	125	313.00
1600	512	116952	10674	3954	123	317.67
800	9188	104947	551	47618	318.5	122.67
800	8192	104888	617	54999	369.5	105.72

800	4096	106193	1239	15071	126.5	309.03
800	2048	107120	2495	14185	335	116.61
800	1024	111869	5095	8485	248.5	157.30
800	512	116651	10640	6140	249.5	156.58
400	9188	-	-	-	-	-
400	8192	-	-	-	-	-
400	4096	11382	137	2692	25	156.25
400	2048	10715	255	2828	70.5	55.48
400	1024	11200	510	1496	56.7	68.89
400	512	11776	1083	1168	50	78.13

Annex D Cell loss localisation

During ATM multiplex experiments unexpected cell loss occurred in case two rather small bursts are multiplexed. Furthermore cell loss occurs in case large bursts are received by a Silicon workstation. Causes for these cell losses can be the FORE ATM cards, the GDC switch or the workstations. Performing some experiments the cause of the cell losses is located.

Cell loss in the GDC switch

It is suspected that the buffer size is not as large as assumed in paragraph 7.3.4. This is verified by a buffer experiment.

When a connection is set up between the Silicon workstations via the GDC switch and Amsterdam (external loop), cells arrive at the GDC switch with 100 Mb/s and depart with 34 Mb/s. So a number of cells has to be temporarily stored in the buffer of the GDC switch. This is depicted in figure .

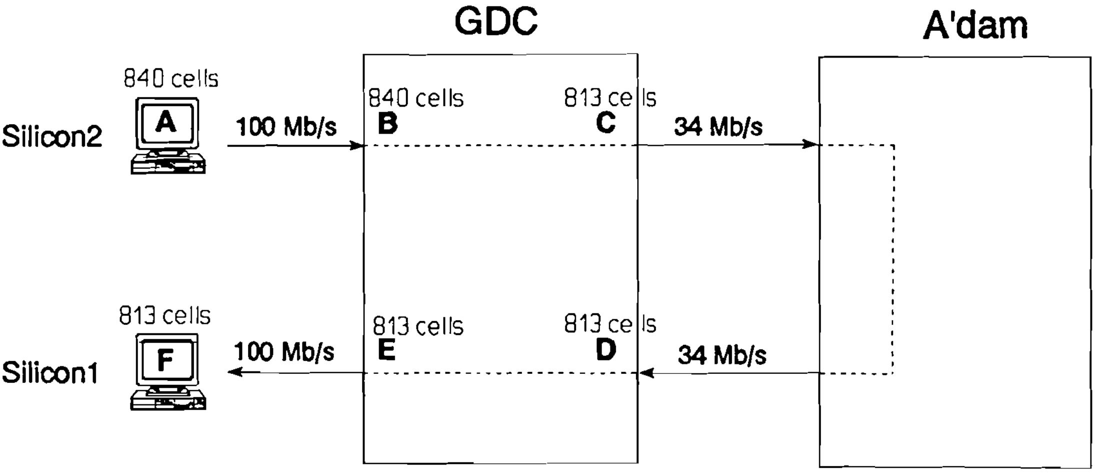


Figure D.1 *Cell loss in the GDC switch ?*

Sending a ping message from Silicon2 to Silicon1 via the GDC switch and the switch in Amsterdam, cell losses occur for ping messages larger than 2650 bytes (56 cells). To locate the cause of these cell losses, a ping message of 2800 bytes (60 cells) is sent at Silicon2 for 14 seconds, so `atmstat` at point A displays 840 cells in total. The GDC switch receives also 840 cells at point B, but only transmits 813 cells at point C. All cells sent to Amsterdam (813 cells) are also received again by the GDC switch at point D and transmitted to Silicon1 at point E.

According to `atmstat` at Silicon1 813 cells are received.

Only 1 segment is received complete (60 cells) and an ACK of 60 cells is sent back to Silicon2 (point F). These 60 cells are received by the GDC switch (point E), but only 57 cells are transmitted at point D. In Amsterdam no cells get lost and 57 cells are received at point C, point B and according to `atmstat` Silicon2 receives 57 cells, but these are dropped since the segment is not complete.
This experiment is repeated using Silicon2 as transmitter and EMMA2 as receiver and vice versa. Also the same experiment is performed between the EMMA workstations. In all cases the outcome was identical to the results above.

It can be concluded that buffer overflow in the GDC switch is the cause of the cell loss. It was assumed that the GDC switch contains two independent output buffers of 63 or 1171 cells and that the large buffer was used for these experiments.
However, this experiment proves that the real buffer in the switch is a lot smaller. During other (throughput) experiments ([Wal95]), it appeared that the Silicon workstations produce a burst of cells at a rate of about 75 Mb/s.
Using a fluid flow approximation of the arrival and departure process, the size of the buffer can be calculated:

$$\text{buffer size} = \frac{\text{arrival rate - departure rate}}{\text{arrival rate}} \text{ x burst length} = \frac{75 - 34}{75} \text{ x } 56 = 31 \text{ cells.}$$

This complies with the GDC manual that later became available and mentions that the C series cards employ a single 31 deep output buffer, while the H series cards has two independent output buffers as was assumed before. So apparently the C series cards are implemented in the GDC switch at the moment. For most experiments with the Silicons and for multiplex experiments, this output buffer is too small and therefore it is desirable to implement the H series cards soon.

Cell loss at the FORE 100 ATM cards for large bursts

In case a connection is set up between the Silicon workstations via the GDC switch (internal loop) and a ping of 26000 bytes or more is sent from Silicon2 to Silicon1 cell loss occurs. The question is where these cells get lost; at the transmitter side already, in the GDC switch or at the receiver side. Using the `atmstat` command, the number of cells at input and output is displayed (point A and D). The number of cells received and transmitted by the GDC switch is measured at the GDC Manager (point B and C).

The cause of the cell loss is located, performing a simple experiment (see figure).
First a connection is set up between the Silicon workstations. The command `ping -s 28000 silicon1` results in 589 cells. So after 7 seconds 4123 cells are transmitted (point A). At the input of the GDC switch (point B) also 4123 cells are received and at the output of the switch (point C) 4123 cells are transmitted. However only 3570 cells are received by Silicon1 (point D). Since not a single complete ping message is received by Silicon1, no cells are sent back to Silicon2.

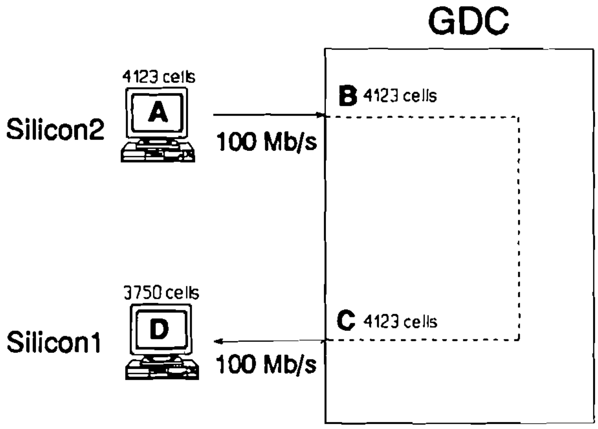


Figure D.2 Ping message from Silicon2 to Silicon1.

Now the transmitter and the receiver are exchanged and the experiment is repeated. Silicon1 sends pings with a size of 28000 bytes for 11 seconds, which results in a total of 11 x 589 = 6479 cells sent (point A). The input side of the GDC switch receives 6479 cells (point B), while the output side transmits 6479 cells (point C) according to the GDC Manager. At the Silicon2 only 5221 cells are received (point D).

Again since not a single complete ping message is received by Silicon2, no cells are sent back to Silicon1. These results are depicted in the following figure.

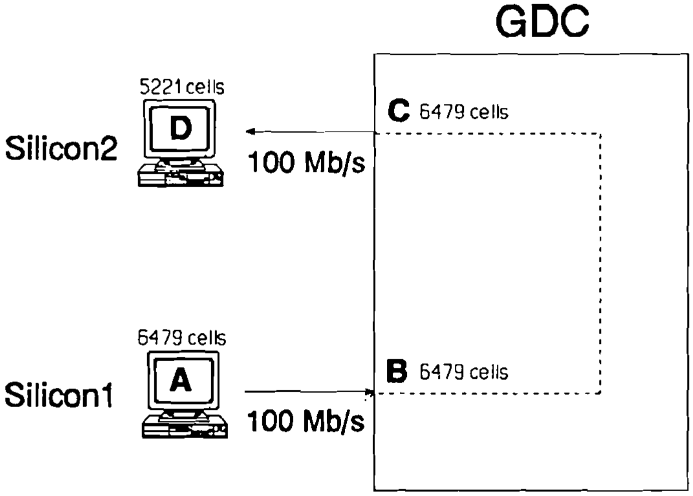


Figure D.3 Ping message from Silicon1 to Silicon2.

These experiments indicate that no cells are lost in the GDC switch, but cells get lost between the output of the GDC switch and the receiver. The FORE 100 card is suspected to be the cause of the cell drops.

In case a connection is set up between the Silicon2 and the EMMA2 workstations via the GDC switch, the receiver contains another ATM card (FORE 200). Silicon2 sends pings with a size of 28000 bytes for 7 seconds to EMMA2, which results in a total of 7 x 589 = 4123 cells (point A). The input side of the GDC switch receives 4123 cells (point B), while the output side transmits 4123 cells (point C) according to the GDC Manager. At the EMMA2 4123 cells are received (point D), see left side of figure D.4.

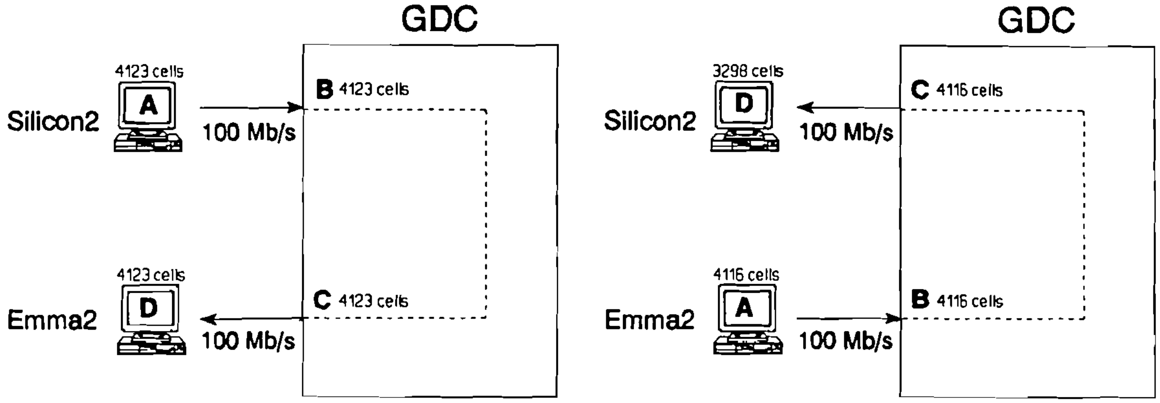


Figure D.4 Left: Ping message from Silicon2 to EMMA2, Right: Return message from EMMA2 to Silicon2.

EMMA2 answers each received ping message by sending a return message of 588 cells. So in total EMMA2 sends 7 x 588 = 4116 cells back to EMMA1 (point A). These cells are received by the GDC switch (point B) and 4116 cells are transmitted towards their destination (point C). But at Silicon2 only 3298 cells are received (point D), see right side of figure D.4. The reason why the EMMA workstation answers a ping message of 589 cells with a message of 588 cells is unknown*.

Unfortunately now the transmitter and receiver could not be exchanged, since the software on the EMMA workstation can not produce a ping with a size of 28000 bytes.

This second experiment indicates that in case a burst of 26000 bytes or more is received by a Silicon workstation with the FORE 100 ATM card cell loss occurs. When the same amount of data is received by the EMMA workstations with the FORE 200 ATM card no cell loss occurs. So apparently the Fore 100 ATM card is the cause of the cell drops. New Fore 200 ATM cards for the Silicons have been ordered already.

* *Possibly due to a different IP implementation that divides the data differently over the IP datagrams.*