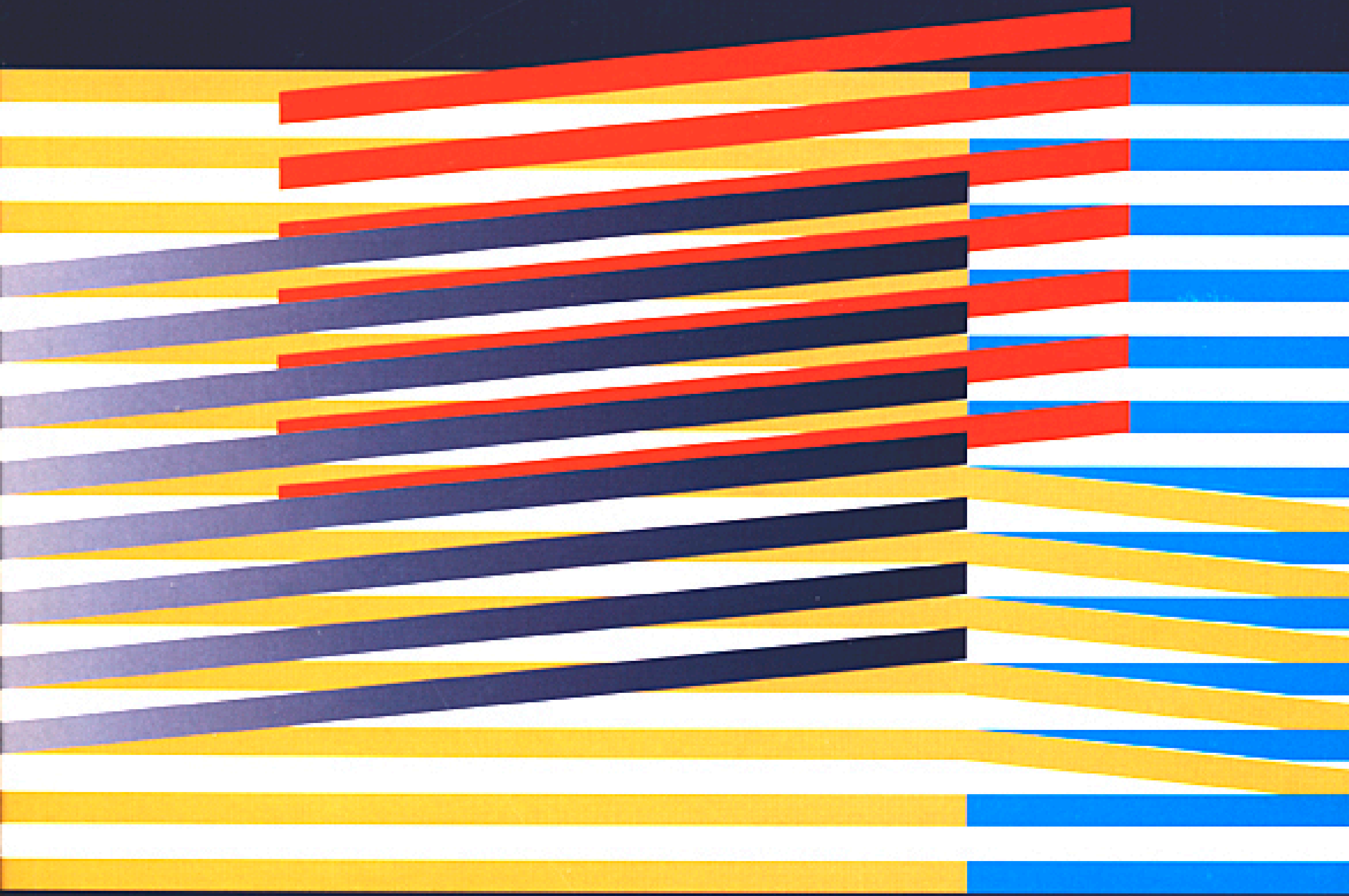
Telektronikk 2/3.95

Teletraffic



Contents

Feature

Status

Telenor

Telektronikk

Volume 91 No. 2/3 - 1995
ISSN 0085-7130

Editor:
Ola Espvik
Tel. + 47 63 84 88 83

Status section editor:
Endre Skolt
Tel. + 47 63 84 87 11

Editorial assistant:
Gunhild Luke
Tel. + 47 63 84 86 52

Editorial office:
Telektronikk
Telenor AS, Telenor Research & Development
P.O. Box 83
N-2007 Kjeller, Norway

Graphic design:
Design Consult AS

Layout and illustrations:
Gunhild Luke, Britt Kjus, Åse Aardal
Telenor Research & Development

Guest editorial

BY ARNE MYSKJA

The situation is familiar: Some nice, new communication system with fancy facilities is installed, and everybody is happy. Until some day the system response gets slow, or blocking occurs more and more frequently. Something must be done, but what? In a simple network it may be a straightforward matter of adding capacity, even though on the way costly time is wasted. In the more complicated systems diagnosis is also more difficult. One may have to do systematic observations and carry out sophisticated analyses. The problem is no longer that of correct operation in accordance with the functional design of the system. It is rather a matter of how to give service to many uncoordinated users simultaneously by means of a system with limited capacity.

With the extremely large and complicated telecommunications networks of today two main considerations may be pointed out: *functionality* and *quality*. An important subset of quality characteristics is that of *traffic performance*. A functionally good solution may at times be rather useless if the traffic dimensioning and control are inadequate. In this issue of "Telektronikk" *teletraffic* is chosen as the theme in focus.

In the early days of telephony – around the last turn of the century – users encountered blocking and waiting situations because of shared subscriber lines, inadequate switchboard or operator capacity, or busy or unavailable called users. Later, trunk lines between switchboards became a concern, and the introduction of automatic switches – for all their advantages – stripped the network of intelligent information and control functions. Many of today's teletraffic issues were in fact present in those early systems: shared media, limited transmission and switching capacities, control system limitations and called side accessibility. Like in the early days, blocking and delays result.

The first systematic studies of teletraffic were carried out about ninety years ago. Several people initiated studies of telephone traffic, using probability theory. However, it was the Danish scientist A.K. Erlang who pioneered a methodical study that is still fully valid. His main publications appeared in the period 1909 – 1926, with the most important contribution in 1917.

The state of the development of teletraffic theory today can be illustrated in several ways. The main forum of contributions is the International Teletraffic Congress (ITC). Since 1955 fourteen congresses have taken place with increasing world-wide participation. Only at the last congress in 1994 more than 1500 publication pages were presented. In addition, an impressive number of regional and national conferences on the subject take place. Many other telecommunications conferences include teletraffic as part of their program, and standards organisations have teletraffic on their agenda. Teletraffic theory is taught in many universities, journal articles abound, and numerous textbooks have appeared. Queuing theory and operations analysis are concepts closely related to teletraffic theory, but distinctions will not be discussed here.

Traffic definition in itself is extremely simple. The instant traffic value at a certain point in a system is simply $A(t) = i(t)$, $0 \leq i \leq n$, where i is the number of occupied servers among n accessible servers at that point. The mean traffic value A over a given interval T is the time integral of $A(t)$ divided by T. Thus, a traffic value is simply given by a number with no denomination. Traffic is created by calls (arrivals) and service times, and the most basic traffic formula is Little's formula: $A = \lambda \cdot s$, where λ is the mean arrival rate in calls per time unit and s is the mean holding time. This formula applies to any part of a system or to the whole system, and it is independent of distributions, so that the single parameter A may often replace the two independent λ and s.

Given the simplicity of concept, why then the virtually endless number of different cases and the complexity of problems? The answer is best given by first assuming the simplest conditions: time-invariant basic process, independence between single events and fully accessible service system. This is one single case, where only a small set of parameters is a matter of choice. However, as soon as one or more of these conditions are dropped, variations are virtually endless. Not only are the cases numerous, also the analyses grow much more complex.

A question sometimes posed is: When electronics and software tend to produce functions of control, switching and transmission at a much lower cost now than earlier, would it be sensible to avoid sophisticated dimensioning and simply add capacity to be on the safe side? I readily admit that I find the question worth considering. Still, the proposition sounds like an echo. At each new major step in the development of telecom networks the focus of performance analysis has shifted. Up till now these shifts have not led to loss of interest in the performance issue. The increasing frequency of and attendance at teletraffic conferences bear witness to the opposite. But there is not only that evidence, there is also good reason behind. Simply trying to guess the needs would in many cases lead to underprovisioning of capacity with initial troubles and costly additions, or otherwise to overdimensioning with unknown amount of unnecessary capital invested. An interesting observation is that overdimensioning very often went undetected since nobody ever complained!

My presumption is that one will always need to understand the mechanisms and to carry out analyses of traffic performance, whatever are the traffic types, the system solutions and the cost of establishment and operation. There are no indications that the costs will be such that decisions should be taken on a basis of guesswork, or even experience of poor functioning. A solid theoretical basis of analysis and dimensioning, updated to cover the system state of the art, will always be necessary.

It is not the ambition to cover every aspect of teletraffic in the present issue of “Telektronikk”. The Norwegian base is deliberately emphasised by inviting mainly national authors. Thus, the colouring of the present issue is very much given by the present traffic oriented activities in Norway. The limitations of this are very clear, since there has to be a rather modest number of traffic specialists in a small country with no dominant telecom industry. Still, there are very capable people who did not have the opportunity to participate on this occasion. In view of the good Scandinavian co-operation, primarily through the regular Nordic Teletraffic Seminars, the scope is extended to a very limited number of contributions from our neighbours. Many more would be desirable. As is well known, the Scandinavian contributions to teletraffic theory and applications have been very substantial.

As the guest editor of the present journal issue I was asked by the chief editor to produce an extensive introduction to the main subject. The introduction ought to be readable by non-experts in teletraffic matters and in the more demanding mathematics. The result appears on the following pages. In view of the fundamental importance of A.K. Erlang’s works, a reprint of his historic paper from 1917 in English translation, along with a brief introduction, is included. Another paper, by R.B. Haugen, is dedicated to the Scandinavian pioneer Conny Palm.

The more general concept of quality of service is approached by F.A. Aagesen. He points out that the quality of service often has got less attention than the functional properties, and more so in connection with data packet networks than in traditional telephone networks. A general QoS approach related to the OSI model is discussed. The results of very extensive traffic measurements in Finland are presented by A. Parviala, showing among other that traffic tends to vary more over time than what is often assumed in dimensioning practice. B. Wallstrøm has submitted a new extension to the equivalent random theory (ERT) for point to point blocking calculations in a hierarchical network. U. Kørner discusses overload conditions in common control equipment, and G. Nossum & al present structure and dimensioning principles for the S12 system with emphasis on remote ISDN units. Structure, routing and dimensioning principles in a new target network based on SDH with mesh and ring topologies are presented by A. Østlie. T. Jensen is responsible for the only contribution on performance analysis and simulation of mobile systems.

A traditional set of observations on a large digital PBX, where data collected by charging equipment is the main source of traffic observations, is presented by B. Feng & al, and an example of traffic measurements in 14 different local area networks (LANs) is reported by S. Gaaren.

It must be expected that a high proportion of the contributions would centre around the hot area of high speed (broadband) communication. Already the mentioned measurements of LANs point in that direction. No less than 7 of the papers are directly related to the asynchronous transfer mode (ATM). A survey of analytical methods in the study of ATM is offered by I. Svinnset, and S.O. Groven discusses objectives and methods regarding measurements on ATM switches. Even some initial measurement results are reported, focused on cell loss, errors, delay and delay variation. K. Moldeklev & al present throughput observations with TCP/IP transmissions over ATM. The requirements regarding access control, largely determined by the diversity of capacity needs of multiservice networks, is dealt with in the paper by H. Pettersen & al. Experimental results are included. A comprehensive study of the traffic generation process is reported by B. Helvik, along with a description of a synthesised traffic generator for ATM traffic. The appearance of very infrequent significant events in ATM transmission and switching is the background for a study on speed-up techniques in simulation, reported by P. Heegaard.

With the stated intention of keeping mathematics at a reasonable level in spite of its importance in traffic analysis, I am pleased to note that we have succeeded to a reasonable extent. We must, however, recognise the need for more sophisticated methods. It is only fair to include some illustration of the high complexity inherent in many traffic problems. An example of complex mathematical modelling is offered by O. Østerbø in a study of queuing models for ATM. Also the paper by E. Jensen on processor performance in call processing within contemporary switching systems is demanding in its mathematical formulation.

Before closing the edition of this special issue, as a guest editor I want to express my appreciation to all those involved. First of all this goes to the authors, who so enthusiastically participated in the efforts. The very competent editorial staff at TF (Telenor Research) has carried out the chore of details, and even added the more artistic traits. I thank chief editor Ola Espvik, who incited me to accept the task, and who has given me a free hand, still giving all the support asked for.

An introduction to teletraffic

BY ARNE MYSKJA

1 Introduction

This article is intended to be a broad introduction to the subject of teletraffic, particularly written for a special teletraffic issue of the journal *Telektronikk.* The presumed readers are primarily telecommunications engineers, but also others who have an interest in the subject of teletraffic, without being – or wanting to be – an expert in that particular area. Traffic theory is covered by numerous textbooks as well as by an impressive amount of papers found foremost in the records of ITC (International Teletraffic Congress). A particular support has been the textbook "Data- og teletrafikteori" [1] by Villy Bæk Iversen.

With this stated intention it might seem an easier task, since the burden of mathematical rigor to some extent is relieved. On the other hand, the development of mathematical theory for the purpose of modelling, dimensioning and optimisation of telecommunications systems has turned out to be among the most powerful tools available. A non-mathematical description can never catch some of the most important aspects of the task. The question I have posed is: Can one make easy reading out of a complicated matter? My pragmatic choice is to aim at simple textual explanations and to use fairly simple mathematics supplemented by illustrations in the forms of functional diagrams, curves, tables, etc. Some elaboration on distributions, as well as deduction of formulas, have been put into "boxes" that can be studied separately. Even there the reader will find little use of "higher" mathematics. Basic arithmetic is supplemented by simple integrals, exponential functions and a limited use of Laplace- and Z-transforms. Some readers may be unfamiliar with – and even a little scared by – transforms. I want to stress the simplicity of concept and the usefulness of those transforms.

The presentation is coloured by my particular interests. Thus, my background in telephone traffic measurements as well as modelling by moment matching and repeated calls studies has certainly had its influence. Still, I hope the general view of traffic is predominant. There is, of course, the risk that many knowledgeable people will find the presentation trivial, since they already know much more about traffic. I do apologise, and suggest that they only browse through this introduction and rather concentrate on the more specific articles in the issue.

An obvious cause of criticism will be that of the length of the article. Who will read one article of about 30 journal pages? In fact I was incited by the chief editor to attempt to write a comprehensive introduction of such extent. I assume my target readers to belong to some of the following categories:

- the "browse through" experts
- those who look for basic formulas
- those who look for development of basic formulas
- those who want to study particular sections in more detail
- those who want to read the complete text as a condensed book.

I wish all my readers a pleasant journey, whether it is an initial tour, or it is the n^{th} repetition.

2 What is teletraffic?

The question is often posed by students at the beginning of a course in the subject. Before any mathematical definition is given it may be useful to seek an intuitive one.

Traffic or *traffic intensity* is a non-denominate and non-physical measure of *load* on a system. It is thus given by a *pure number* with no physical unit attached to it. The load is simply a zero/one matter of a *server* being free/occupied. A server may be any type of resource entity that has this dual property (line, trunk, switch inlet or outlet, signal receiver, radio channel, memory access, etc.). Because of the importance of traffic as a concept, however, it has been decided to use the notation *Erlang* as a traffic unit. Thus a single server carries a traffic of 1 Erlang if it is continuously occupied. Two servers with occupation 1/4 and 3/4 of the time also together carry 1 Erlang. Traffic is normally related to a *traffic carrying system,* consisting of a discrete number of servers. Each of the servers can at any moment carry a load of one or zero. A system of n servers can carry an instantaneous load A of any integer number $0 \leq A \leq n$. In this sense A is always an integer (discrete) number. The definition implies that two servers of different capacity (say one line of 9.6 kb/s and one of 64 kb/s) both carry 1 Erlang as long as they are occupied to their full capacity, even though the amounts of data transmitted during the same time interval are very different. The capacity matter will be discussed in relation with data communication and various bit rates.

In general practice A is considered an average number over a given time interval. This average will in general be a non-integer (continuous) value. When needed, it should be stated whether traffic means an *instantaneous* value or an *average* one. In the latter case much more specification about the way the traffic varies may be necessary. The number of servers, n, on the other hand is an integer.

The load of a traffic carrying system has to be generated by some set of *traffic sources.* In general, the traffic sources are individuals, customers, that ask for service in a co-ordinated or rather an uncoordinated manner. A request for service is a *call attempt,* which, if granted, will occupy one server as a *call.* Often the term call is used as a synonym for call attempt, when no ambiguity arises. (See list of terms below.) In general an arbitrary number of servers may be requested by one call, or a particular server type of a certain capacity. When nothing particular is said, these options are not considered. Thus, the individual sources are all of the same type, and they will occupy only one server at a time. However, their request for service (average) may vary widely between 0 and 1. The number, N, of sources requesting service from a number n of servers may be any integer $0 \leq N \leq \infty$. It is obvious that always $A \leq N$. Any call will have some *destination.* From a traffic point of view it may be non-addressed (broadcast), multi-address (several addresses), single address multi-server or single address single server.

The term *telecommunication* implies communication over some *distance.* Apart from cases with a permanent source/destination relation, a free selection is assumed. This again implies switching facilities in a network of nodes and links. Basic traffic studies are always related to specific interfaces in the network, be it a link group, switch group or a group of functional equipment. (In more advanced studies a network as a whole may be treated.) While most source/destination traffic in the active state is full duplex and thus non-directional, the establishment of a connection is usually directional, there is an A-side and a B-side. For two reasons there will be a diminishing traffic load in the $A \Rightarrow B$ direction:

1 The delay along the path leads to diminishing holding time

2 Calls are aborted along the path for some reason.

Thus the traffic contribution along the path from a set A of sources to a set B of destinations tends to the relation

$$A_{source} > A_{line} > A_{switch} > A_{link} > \ldots$$
$$> A_{switch} > A_{line} > A_{destination}$$

Improved dimensioning, switching technology, system solutions and signalling systems, as well as automatic answering devices at the destination side, all tend to diminish the above differences.

In CCITT recommendation E.600 88 traffic related terms with definitions are listed. With reference to that list we present just a few of particular interest at this stage, with somewhat incomplete definitions:

- **call** – a generic term related to establishment, utilisation and release of a connection
- **call intent** – the desire to establish a connection (may be suppressed by low service expectations)

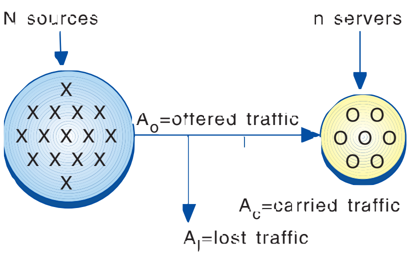


Figure 1 A simple traffic model for lost calls

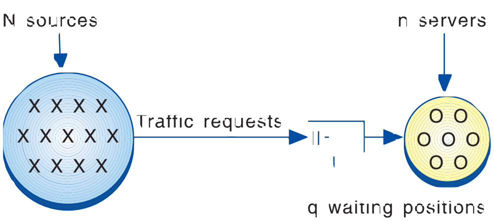


Figure 2 A simple traffic model for waiting calls

- **call demand** – a call intent that results in a first call attempt
- **call attempt** – an attempt to achieve a connection
- **first call attempt** – the first attempt of a call demand
- **repeated attempt, reattempt** – any attempt after the first, relating to a demand
- **call string** – all call attempts related to a single demand.

The distinction between call intent and call demand is worth noting. In a network of poor quality there may exist a "hidden" traffic demand that will only be manifested after a distinctive upgrading of the network service quality. There are many examples of a strong traffic growth after replacement of outdated equipment by better dimensioned and technically improved system.

3 A lost call traffic model

Up till now we have only discussed traffic A as a measurable load on a set of servers or an occupation state of a set of sources or destinations. If $N > n$ a new call for service may arrive while all n servers are occupied. The new call will not get immediate access to a server, and if the call cannot be put on waiting, the call is lost and will not lead to any traffic load. However, the lost call represents a *traffic potential* that would have been realised if there had been at least one more server available. Thus it is clear that if $n \geq N$, the full potential will be realised. This represents a traffic offer, part of which may be lost when $n < N$. The assumption is now that a lost call would on average contribute the same amount of traffic volume as a carried call, and we can define three traffic amounts:

A_c = carried traffic (real, measurable)

A_l = lost traffic (fictitious, non-measurable)

A_o = offered traffic (partly real, partly fictitious, non-measurable).

By definition we have

$$A_c = A_o - A_l$$

A simple model is shown in Figure 1.

The traffic carried by a server being part of an extended network consists of two periods, before and after the reply. The former period represents a *set-up* traffic and the latter a *conversational* traffic, also called *effective* (completed) traffic A_e. From the network point of view one might contend that a call is completed by the time the alerting (ringing) signal is sent to the called part, and this is also used as a criterion of the network efficiency. As indicated above, the non-effective (set-up) traffic will be decreasing along the path from calling to called part. In the more modern systems the non-effective traffic is reduced, so that in the end virtually only dialling (addressing) and alerting times remain.

4 A waiting call traffic model

The assumption that a call request is not immediately lost when all servers are occupied, but put in a waiting position, changes the model to the one shown in Figure 2. The simplest assumption is that the number q of waiting positions is such that $q + n \geq N$ and that the sources have unlimited patience.

With a limited number n of servers the above condition implies that if $N \to \infty$, then $q \to \infty$, and there is an unlimited number of traffic sources and waiting positions. The model can be modified to have sources with limited patience and/or $q + n < N$, in which cases we have a combined waiting and loss system.

5 Traffic as a process in time and space

At this stage we have already referred to a number or a count of entities, like servers, waiting positions and sources, and some time reference. In that scenario we may consider the traffic as a *process* with the two dimensions, *space* (number of occupied entities) and *time* (instant, duration). In principle, the traffic may be described in relation to *discrete* or *continuous space* and *discrete* or *continuous time,* which gives altogether four modes, Table 1.

Table 1 Four traffic modes in time and space

Space \ Time	Discrete	Continuous
Discrete	(x)	x
Continuous	((x))	((x))

As already indicated, continuous space is not very interesting in teletraffic, but it is sometimes used in mathematical analysis, when traffic is considered as a diffusion process, often with rediscretisation as a final step. Also in intermediate analysis stages, calculation of fictitious continuous server groups may be used to obtain improved accuracy. Time is most often considered to be continuous, as the processes studied are often defined by random events. However, in modern digital systems synchronous operation may be better described in discrete time. If nothing particular is indicated, discrete space and continuous time is assumed. This means that at any instant an *integer* number of servers are occupied, and that the number may change at any instant in *continuous* time. In a so-called *orderly* process (to be discussed later), any change is by ±1 only.

6 Traffic variations

6.1 Telephone traffic

The traffic load on a traffic carrying system is subject to more or less regular variations on different time scales. In order to understand the causes behind such variations one has to study the natural activity patterns of the set of traffic sources and the resulting behaviour in relation to the system. This behaviour is not independent of the system response.

One can imagine a basic traffic potential under given conditions of a well dimensioned system and reasonable tariffs. The system carries that potential, and it can be said that the system feedback is weak. If for some reason a narrow bottleneck occurs, many call attempts fail. The result is double: 1) some failed attempts are repeated, thus increasing the call rate and giving a further rise in the lost call rate, and 2) some failed attempts lead to abandonment, thus leading to less carried useful traffic. (These effects will be discussed later under repeated calls.) In general an improved system, as sensed by the users, and cheaper tariffs are also feedback that tend to increase the traffic. In this section we will assume well dimensioned system and stable tariffs.

We assume a set *(N)* of traffic sources that operate independently. This is a realistic assumption under normal circumstances. The service system consists of a set of servers *(n)*, where *n* is large enough to carry the traffic demand at any time. A time function of the number of busy servers depicts the stochastic variation of the carried traffic. It is a discontinuous curve with steps of ±1 occurring at irregular intervals, as shown in Figure 3.

An observation period *T* is indicated, and it is possible to define an average traffic $A_m(T)$. At this point it is convenient to define a *traffic volume,* being the integral under the traffic curve

$$V(T) = \int_o^T r(t)dt \tag{1}$$

where *r(t)* is the number of busy servers (the instantaneous traffic value) at time *t*. The mean traffic value is given by

$$A_m(T) = \frac{V(T)}{T} \tag{2}$$

A 24-hour integration gives the total traffic volume of that period, and hence expresses the economic value of the carried traffic. However, it is a poor indication of the need of traffic carrying capacity. On the other hand, a dimensioning that allows any instantaneous peak value to be carried, cannot be defended for economic reasons.

The diagram of Figure 3 gives the full detail. If a good stochastic model is identified, it is better to work with averages over defined periods and use that stochastic model on top of the average. One possibility is a sliding one hour average, which gives a nearly continuous curve like the one in Figure 4, which allows to pick exactly the highest one hour period of the day. An alternative diagram (not shown) is found by picking the 24 one hour interval columns. The standardised method is based on quarter-

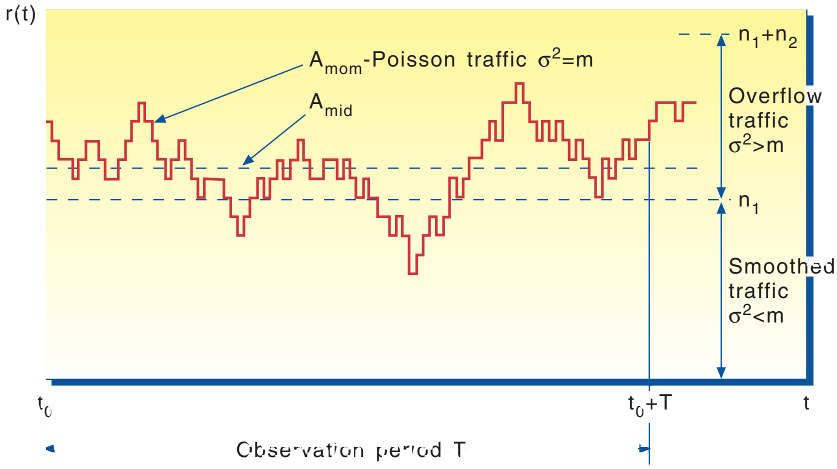


Figure 3 Stochastic traffic variations over an observation period T

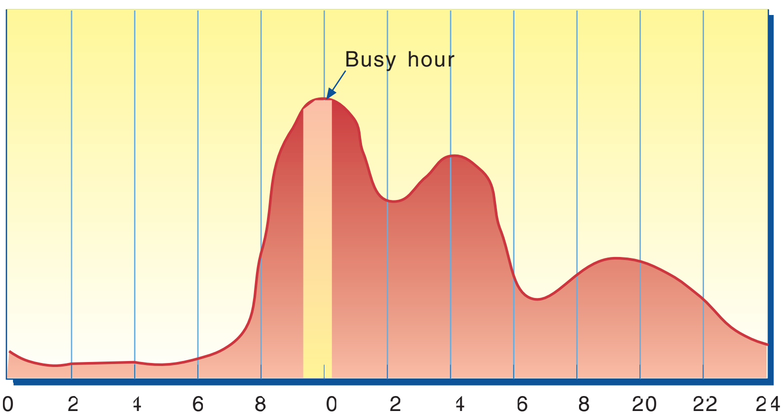


Figure 4 Typical 24-hour traffic variation

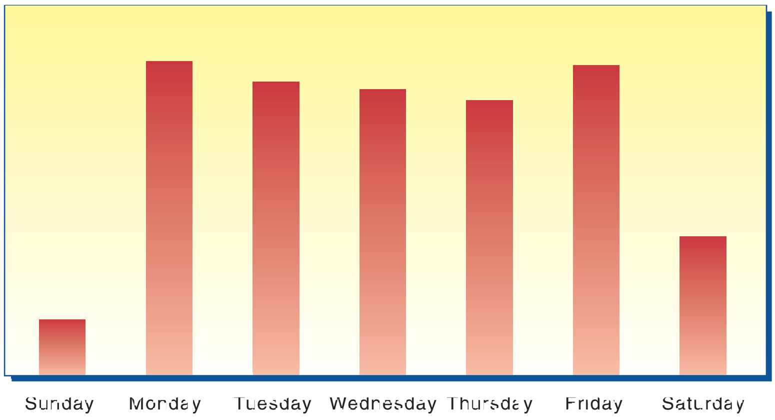


Figure 5 Typical busy hour traffic variation over seven weekdays

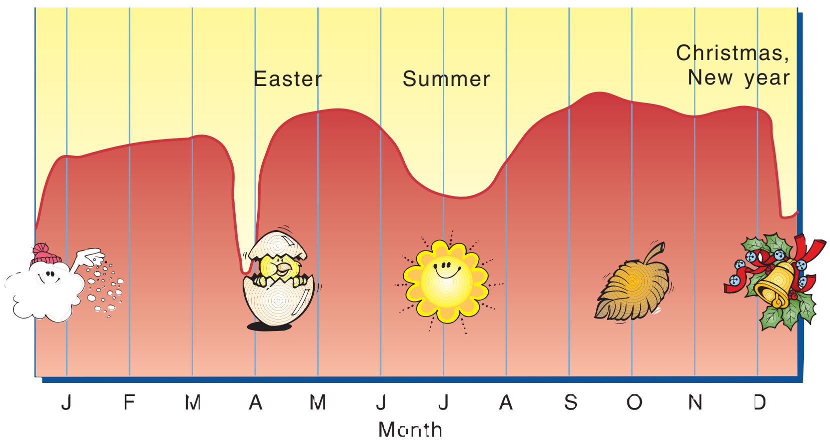


Figure 6 Typical busy hour variation over one year

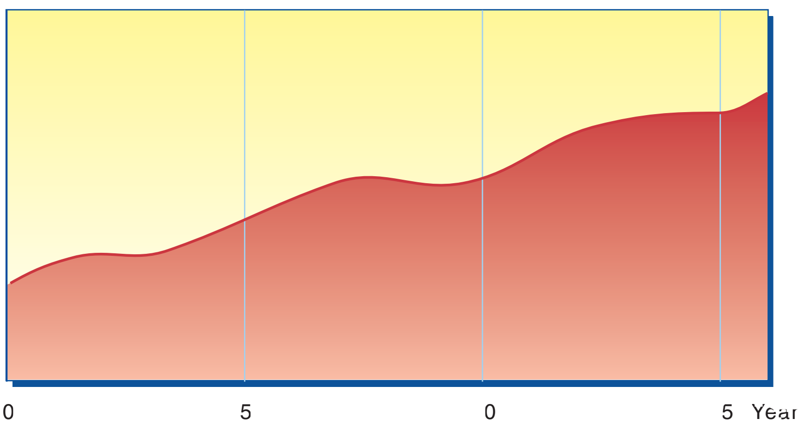


Figure 7 Many-year traffic trend with market fluctuations

hour measurements, and on picking the four consecutive measurements with the highest sum, which will give results close to the sliding average method. This can be done for each day, or for the ten working days of a selected two-week period. (See the discussion in [2], this issue.)

The diagram of Figure 4 or similar can be used to determine the *busy hour*. The busy hour is used as a dimensioning basis. In the following diagrams the daily busy hour is assumed as the basis. Figures 5–7 indicate variations over one week, one year and a many-year period. All the diagrams are assumptions, showing known typical variations. As pointed out in [2], real traffic measurements indicate less regularity on all time scales than what is usually assumed.

The busy hour is not an unambiguous term. In an unqualified statement it should be defined for a single day. Because of great fluctuations over time, traffic dimensioning and management must consider a much wider basis. Thus, three main concepts have been defined in CCITT Rec. E. 600 (definitions here abbreviated):

- **busy hour** – the continuous 1-hour period for which the traffic or the number of call attempts is greatest
- **average daily peak hour (ADPH) traffic** – the average busy hour traffic of several days
- **time consistent busy hour (TCBH)** – the 1-hour period starting at the same time each day for which the average traffic is greatest.

The busy hour may, as seen, be defined according to traffic or to call attempts. For measurement ADPH or TCBH require continuous recording over long periods. A simpler alternative is to estimate a likely busy hour and do one hour measurements at that time of each day, Fixed Daily Measurement Hour (FDMH). This is, however, a very uncertain method. In practice ADPH measurements are simplified to cover full hour or quarter hour periods (ADPFH or ADPQH) (CCITT Rec. E.500).

It is worth noting that in any observation series like the one in Figure 3, showing stochastic variations around a mean A_m, this mean in itself is a stochastic variable. The determination of a true long-term mean would require very long measurement series. However, in real life such a long-term mean cannot be sustained, since the real variations are not just random variations around an invariant mean. The mean itself varies widely in a non-controllable way. Even during a single busy hour an invariant mean cannot always be assumed. A simple underlying fact is that the number of active sources varies, along with the activity pattern of each source. This may be so even when there is no dependence between any of the sources.

The various profiles (apart from the short term stochastic profile) show the rhythm of activities among the telecommunication users. That applies to the profile of day/night, week and year. Profiles are different for different types of user groups, notably between business users and residential users. Those differences are clearly mirrored in profiles for exchanges and trunk groups with dominant populations of one or the other type. Also, vacation areas may have high season when business areas are at their low. A particular pattern is found for traffic between different time zones. The union of office hours for two areas decreases with time zone difference, and that tends to compress the traffic in a correspondingly short period per day; similarly for residential traffic, but with different hours.

With limitations in the system, further variations are caused by feedback. Particular external events, predictive and non-predictive, also influence the traffic pattern. The predictive events may be regular, like Mother's Day or Christmas Day, or they may be scheduled, like a World Championship or a European song contest. Non-predictive events are above all natural disasters like hurricanes, volcanic eruptions or earthquakes.

It may be obvious that the design of traffic carrying systems must take into account all the different variations. This is done first of all by dimensioning rules based on thorough knowledge of the regular behaviour, balancing the cost of investments and operation against the cost of traffic loss and delay. Secondly, due consideration of irregular conditions requires system robustness and managerial measures to keep the system in an efficient state. We shall come back to the matter of dimensioning and control. At this stage we shall simply sum up the types of variations:

- Short and long term stochastic variations
- Regular 24-hour profile
- Regular week profile

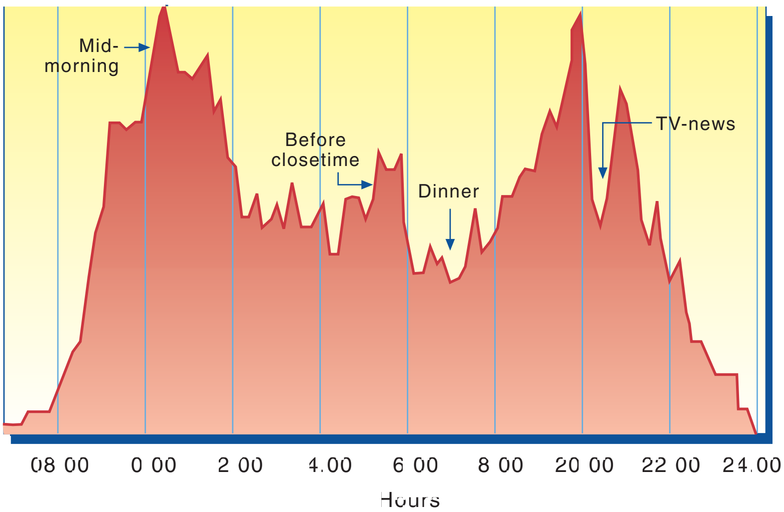


Figure 8 Traffic profile with 10 minute averages for residential subscribers

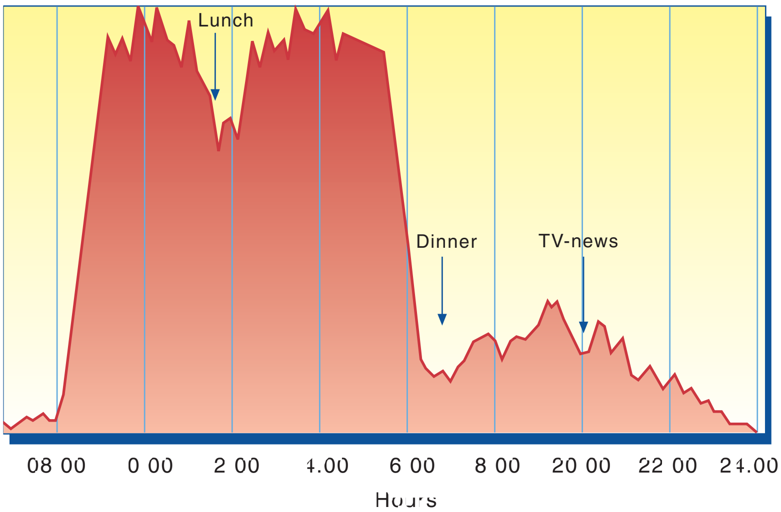


Figure 9 Traffic profile with 10 minute averages for business and residential subscribers combined

- Regular year profile
- Trend
- Market influenced fluctuations
- Variations caused by feedback
- Variations caused by external events, predictive, regular
- Variations caused by external events, predictive, irregular
- Variations caused by external events, non-predictive.

The short term stochastic variations are assumed to be on top of all other variations. The long term stochastic variations influence the regular variations to make them less regular.

6.2 Non-telephone traffic

The general characteristics of traffic above do not presume any particular type of traffic. However, the examples of variations are typical for telephone traffic. The particular properties of such traffic are related to the conversational dialogue in real time between personal users. Other communication types may have quite different characteristics:

A dialogue between a person at a terminal and a computer is inherently non-symmetric. The personal user sets up messages that are transmitted in short bursts with rather long intervals. The computer reply may be very short to very long. File transfers cover a very wide range in bit rates as well as duration.

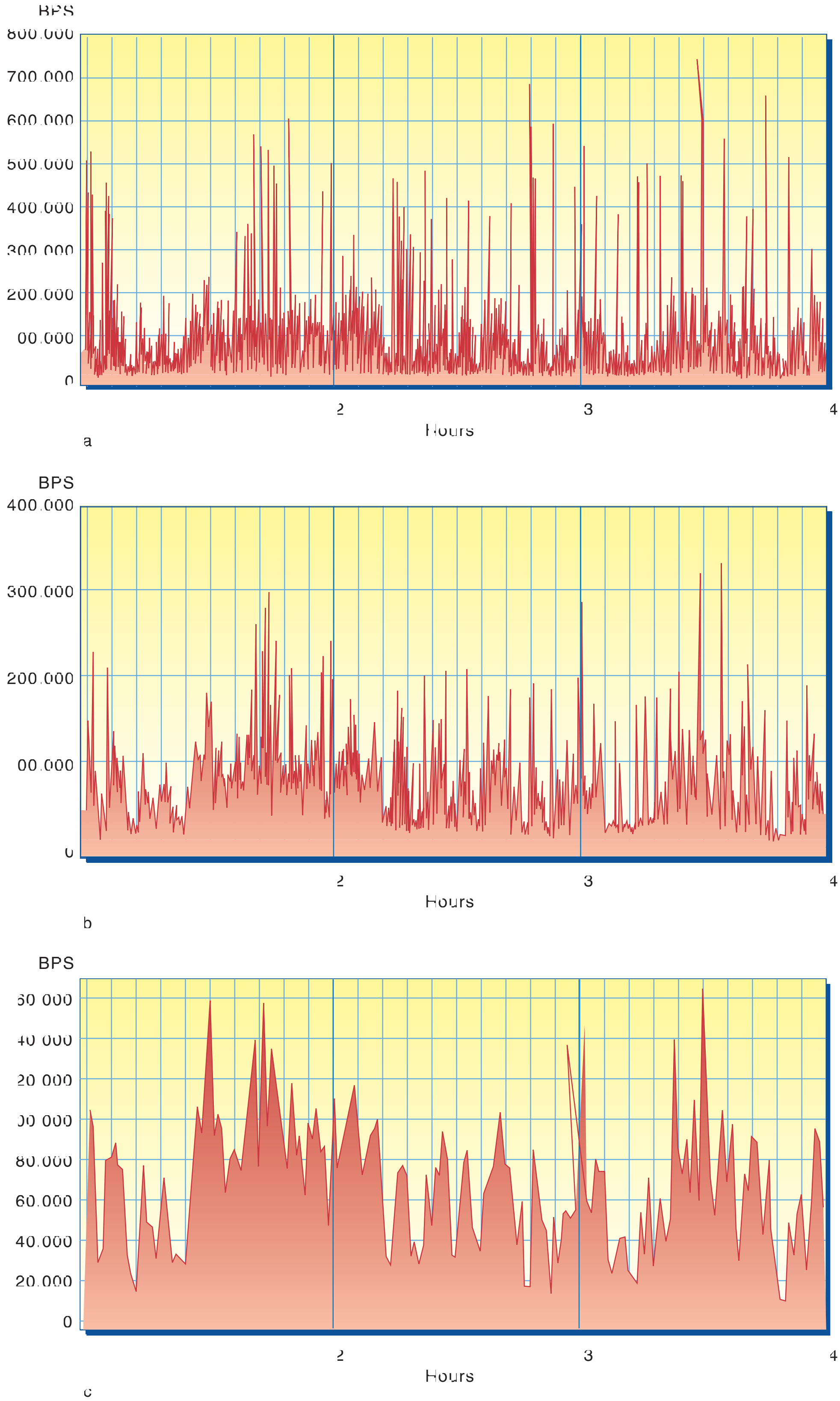


Figure 10 Time variation of a data sample with different integration times
a) 1 second integration time
b) 10 seconds integration time
c) 1 minute integration time.

Backup transfers are often programmed to be performed at night time. Video transmissions are usually one-way, with long-lasting high bitrate transmission, and tend to be mostly evening entertainment. The transmission speed of speech is normalised to a fixed bit rate, whereas data traffic bit rates vary over an extremely broad range.

These and other peculiarities may lead to stochastic variations as well as short- and long-time profiles that are quite different from those of telephone traffic. However, the activities of business hours and leisure hours are in many ways decisive for profiles even for other types of traffic than telephone. And the basic laws of the traffic theory are invariant, while the environments, the conditions and the relevant parameters are highly variable. If the independence assumption no longer applies, correlation must be considered, and the analysis is much more complicated.

Some diagrams showing real traffic observations are presented in the following section.

7 Some traffic observations

As an illustration two diagrams of real traffic as observed at the main telephone exchange in central Trondheim are shown in Figures 8 and 9. Both figures present daily profiles of 10 minute averages between hours 7 and 24, and averaged over several days in order to get smoothed samples demonstrating typical features. Figure 8 contains the results of 25,000 calls from typical residential subscribers. The main features are a slow growth from virtually zero between 7 and 8 hours, rising to a morning peak around 10 (maximum activity), a fall off until a lower peak between 15 and 16 (towards end of working hours), a marked low around 17 (supper, nap), before increasing again to an evening peak (leisure time) of the same size as the morning peak. The further gradual fall-off until midnight is broken by a sharp drop around the evening news.

The corresponding profile in Figure 9 contains 48,000 calls from business and residential telephones as well. Here, business telephones dominate during working hours and residential telephones during the evening. The main difference is found between 10 and 16 hours, showing a marked drop during lunch hour before an afternoon peak between 13 and 14 of

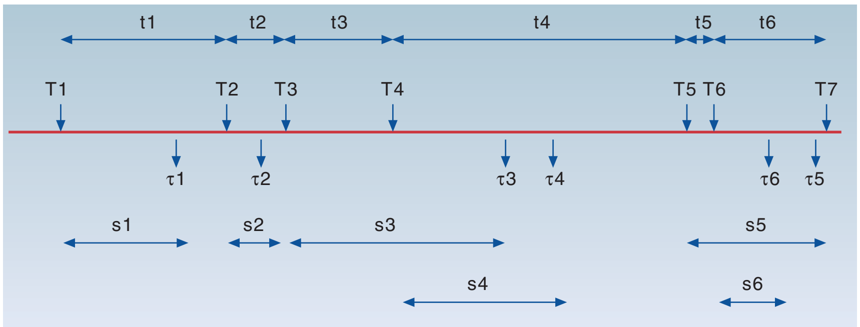


Figure 11 The double process of arrivals and departures

equal size as the one around 10. One observation is that the residential subscribers cause a better utilisation of the system than do the business subscribers, due to the greater traffic concentration of the latter.

An example of data traffic observations [3] is shown in Figure 10, a), b), and c). The observations are given in the form of bits per second from a workstation over a three hour period. Three degrees of resolution are given, 1 second, 10 seconds and 1 minute, clearly indicating the loss of detail by integration over longer time. Indicative is the reduction of peak values from 750 via 350 to 160 kb/s for the same set of data. (Note different ordinate scales.)

8 Traffic modelling

Up till now we have focused on the traffic load on a set of servers and on different types of variation. Apart from an indication that the traffic is generated by calls from traffic sources we have not studied how traffic is created.

In Figure 3 the call arrivals create all the +1 steps. In the long run there are equally many -1 steps, and these all stem from departures from the system. Earlier or later any arrival must result in a departure. This double process may be depicted as in Figure 11. If we start by observing only arrivals, we have the points T_1 to T_7 with the intervals t_1 to t_6.

Arrivals and departures are always connected in pairs. This implies that the long-term averages of arrival rate λ and departure rate γ must be equal. That again secures a limitation of the traffic load as long as λ and the holding times (service times s_i) are limited. It is seen from the figure that the traffic measured during the period $T = T_7 - T_1$ is a little less than 1 Erlang. This is seen by adding the holding times and dividing the sum by T: $A = (s1 + s2 + s3 + s4 + s5 + s6)/T$. By reducing the intervals $T_i - T_{i-1}$, thereby increasing λ, the traffic increases. The same effect is obtained by increasing s_i.

9 Little's formula, the "Ohm's law" of teletraffic

It is well known from electrophysics that the three quantities *voltage v, current i* and *resistance r* are connected by the formula $v = i \cdot r$, and that in an electric network this formula may be applied to the whole network or any part of it. This is Ohm's law. In a similar way the three quantities *traffic A, arrival rate* λ and *service (holding) time s* are connected by the formula

$$A = \lambda \cdot s \qquad (3)$$

This is Little's formula, and like Ohm's law in an electric network, Little's formula applies to the whole or any part of a traffic network. A difference is that for Ohm's law constant values are assumed, whereas Little's formula applies to mean values, including the constant case (given co-ordination of arrivals and departures, which is a bit artificial). The only condition is a *stationary process,* which will be discussed later. In simple terms stationarity means that the statistical properties of the involved processes remain unchanged over time. There are no other conditions laid on the statistical distributions of the variables.

With reference to the traffic models of Figures 1 and 2 an alternative model is shown in Figure 12.

The following indexes can be used:

$o \Rightarrow$ offered, relating to sources
$l \Rightarrow$ lost
$q \Rightarrow$ queue
$c \Rightarrow$ carried, relating to servers.

One could also use corresponding indexing on holding times, however we use, according to common practice, the following:

$s \Rightarrow$ service time
$w \Rightarrow$ waiting time (in queue).

Very often h is used for holding time. It can be indexed to show which system part it is related to. For instance h_l may be used to indicate that even a lost call occupies a source for a non-zero time.

According to Little's formula we obtain:

Traffic load on queue:
$A_q = \lambda_q \cdot w = \lambda_c \cdot w$

Traffic load on server group:
$A_c = \lambda_c \cdot s$

Traffic load on source group (encompassing the whole system):
$$A_o = (\lambda_o - \lambda_l) \cdot (w + s) + \lambda_l \cdot h_l = \lambda_c \cdot (w + s) + \lambda_l \cdot h_l$$

We see that, if the lost traffic holding time h_l is zero, the traffic load on the sources, A_o, is actually equal to the sum of the queue traffic and the server traffic. In this case – but not in general – the "non-empty" call rate is identical for all parts of the system, whereas the holding times are different.

This would be different if the calls waiting in queue had a limited patience, so that some of them would leave the queue without being served. That would reduce the mean holding time on the source group and the queue, and the arrival rate on the server group. Thus the load would be reduced on all parts of the system.

The linear property of Little's formula can be expressed by

$$A = A_1 + A_2 = \lambda_1 \cdot s_1 + \lambda_2 \cdot s_2, \qquad (4)$$

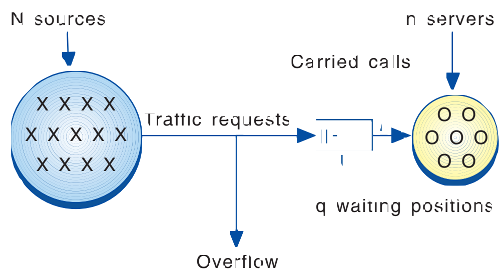


Figure 12 Traffic model with overflow from queue

where the contributions from different traffic streams to a given system part are simply added, irrespective of the distributions of interarrival intervals and of holding times.

10 Traffic-relevant distributions

We assume that basic statistics are well known. In teletraffic processes statistical distributions are very important for modelling and analysing purposes. True processes rarely follow any particular mathematical distribution strictly. Thus, the modelling must be based on approximation.

The approach is to do systematic measurements and to match the results to some known distribution. A primitive way is to look for resemblance to some known distribution and try matching to that one. A more logical way is to use a possible knowledge of the inherent properties of the process and by deduction try to determine the distribution type before doing the matching. An example of the former approach is to take some holding time distribution and from the look of it try matching to exponential, log-normal, Erlangian, Cox or Weibull distributions. The latter – and more satisfactory – approach is for instance to study the incidence of purely random events and by mathematical arguments arrive at the negative exponential distribution. If then the observations agree well, the outcome is very satisfactory.

Frame 1

A condensed section on distributions

General

In traffic theory two main types of distributions are of particular importance:

- Continuous time distributions
- Discrete number (count) distributions.

Continuous distributions

Basic definitions are, X being a random variable:

1 Distribution function (cumulative):

$$F(t) = \int_0^t f(u)du = \int_0^t dF(u) = P\{X \le t\} \qquad (9)$$

2 Survivor function (cumulative):

$$G(t) = 1 - F(t) = P\{X > t\} \qquad (10)$$

3 Frequency (density) function:

$$f(t) = dF(t)/dt = -dG(t)/dt$$
$$f(t) \cdot dt = P\{t < X \le t + dt\} \qquad (11)$$

Only non-negative values are assumed:

$$t \ge 0, f(t) \ge 0$$

For a true distribution we must have:

$$\int_0^\infty f(u)du = F(\infty) = 1$$
$$F(0\_) = 0$$

We may possibly have $F(0) > 0$, with an accumulation of probability in the zero point. (Example: Waiting time having a finite zero probability $P\{W = 0\} > 0$.)

The i^{th} statistical moment of a distribution function may be expressed by

$$E\{X^i\} = M_i = \int_0^\infty t^i f(t)dt = \int_0^\infty it^{i-1}G(t)dt \qquad (12)$$

The mean value or expectation is given by

$$E\{X\} = M_1 = \mu,$$

and the i^{th} central moment by

$$E\left\{(X-\mu)^i\right\} = m_i = \int_0^\infty (t-\mu)^i f(t)dt$$
$$= \sum_{j=0}^{i} (-1)^j \binom{i}{j} M_{i-j}\mu^j; M_0 = 1 \qquad (13)$$

Thus the i^{th} central moment is a linear combination of the i first ordinary moments and a power of the first moment (mean).

The Laplace transform

The Laplace transform is a purely mathematical manipulation implying a two-way one-to-one relationship between a function and its transform. In the present context the main aim of using the transform is to determine the statistical moments of distribution functions. The definition of the Laplace transform is

$$L\{s\} = f^*(s) = \int_0^\infty e^{-st} f(t)dt = \int_0^\infty e^{-st} dF(t) \qquad (14)$$

The second form is also called the Laplace-Stieltjes transform. The L-S-transform of a distribution function is the L-transform of its density function.

The Laplace transform is very useful because it has been solved for a broad range of functions and is easily available from standard handbooks.

The usefulness is illustrated by

$$L\{0\} = \int_0^\infty f(t)dt = M_0 = 1$$

and in general

$$L^{(n)}\{0\} = (-1)^n \int_0^\infty t^n f(t)dt = (-1)^n M_n \qquad (15)$$

Thus the statistical moment of any order n can be found by simply determining the n^{th} derivative and setting $s = 0$. (The negative exponential function in the expression of $L\{s\}$ is sometimes replaced by the positive counterpart, and the transform is denoted the "moment generating function". The term $(-1)^n$ vanishes.)

Frame 2

Useful time distributions

Negative exponential distribution (ned)

The most interesting time distribution is the *exponential* distribution

$$F(t) = 1 - e^{-\lambda t};\ G(t) = e^{-\lambda t};\ f(t) = \lambda e^{-\lambda t} \tag{16}$$

with the Laplace transform of $f(t)$

$$L\{s\} = f^*(s) = \lambda/(\lambda + s) \tag{17}$$

and the n^{th} derivative

$$(-1)^n L^{(n)}\{s\} = \frac{n! \cdot \lambda}{(\lambda+s)^{n+1}};\Rightarrow M_n = \frac{n!}{\lambda^n} \tag{18}$$

A particular property of the exponential distribution is that the coefficient of variation $c = \sigma/\mu = 1$ and the form factor $\varepsilon = v/\mu^2 + 1 = 2$. This is used to distinguish between *steep* ($c < 1$) and *flat* ($c > 1$) distributions. It has a positive skewness $s = 2$.

Hyperexponential distribution

The *hyperexponential* distribution is obtained by drawing at random, with probabilities $p_1, p_2, p_3, \ldots, p_k$, from different exponential distribution with parameters $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_k$

$$F(t) = 1 - \sum_{i=1}^{k} p_i \cdot e^{-\lambda_i t}; f(t) = \sum_{i=1}^{k} p_i \lambda_i e^{-\lambda_i t} \tag{19}$$

For a true distribution we must have

$$\sum_{i=1}^{k} p_i = 1 \tag{20}$$

The n^{th} moment similarly becomes (as the Laplace transform of a sum is the sum of the Laplace transforms)

$$M_n = \sum_{i=1}^{k} p_i \cdot M_{ni} = n! \cdot \sum_{i=1}^{k} p_i / \lambda_i^n \tag{21}$$

Hyperexponential distributions are always flat distributions ($c > 1$).

Erlang-k distribution

If k exponentially distributed phases follow in sequence, the sum distribution is found by convolution of all phases. If all phases have the same parameter (equal means) we obtain the *special Erlang-k* distribution (which is the discrete case of the Γ-distribution),

$$f(t) = \frac{\lambda(\lambda t)^{k-1}}{(k-1)!} e^{-\lambda t} \tag{22}$$

Since the Laplace transform of a convolution is the product of the Laplace transforms of the phases, we have for the special Erlang-k distribution

$$L\{s\} = \left(\frac{\lambda}{\lambda+s}\right)^k \tag{23}$$

and the n^{th} moment is found by

$$(-1)^n \cdot L^{(n)}\{s\} = k \cdot (k+1) \ldots (k+n-1) \cdot \lambda^k/(\lambda+s)^{k+n}$$
$$\Rightarrow M_n = k \cdot (k+1) \ldots (k+n-1)/\lambda^n \tag{24}$$

The mean value is $\mu = k/\lambda$. A normalised mean of $\mu = 1/\lambda$ is obtained by the replacement $\lambda \Rightarrow k\lambda$ or $t \Rightarrow kt$ in the distribution and the ensuing expressions. If $k \to \infty$ for the normalised distribution, all central moments approach zero, and the Erlang-k distribution approaches the deterministic distribution with $\mu = 1/\lambda$.

For an Erlang-k distribution it applies in general that

$$\mu = M_1 = \sum_{i=1}^{k} \mu_i$$

$$m_n = \sum_{i=1}^{k} m_{ni} \text{ for } n = 2 \text{ or } 3$$

$$m_n \neq \sum_{i=1}^{k} m_{ni} \text{ for } n = 4, 5, \ldots$$

For the *general Erlang-k* distribution the phases may have different parameters (different means). The expressions are not quite so simple, whereas the character of the distribution is similar. The particular usefulness of the Erlangian distributions in a traffic scenario is due to the fact that many traffic processes contain a sequence of independent phases.

β-distribution

The *β-distribution* is a two-parameter distribution with a variable range between 0 and 1, $0 \le x \le 1$. Thus, it is useful for description of a population with some criterion within this range, for example average load per destination, urgency of a call demand on a 0 – 1 scale, etc. The distribution density function is

$$f(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\cdot\Gamma(\beta)} x^{\alpha-1} \cdot (1-x)^{\beta-1}; \alpha, \beta > 0 \tag{25}$$

The n^{th} moment is given by

$$M_n = \prod_{i=0}^{n-1} \frac{(\alpha+i)}{(\alpha+\beta+i)} \tag{26}$$

Typical of the above distributions is

Special Erlang-k:	$c = 1/\sqrt{k} < 1$ (steep, serial)
Exponential (ned):	$c = 1$
Hyperexponential:	$c > 1$ (flat, parallel)
β-distribution:	$c > 1$ for $\alpha < 1$ and $\beta > \alpha(\alpha+1)/(1-\alpha)$ $c < 1$ otherwise

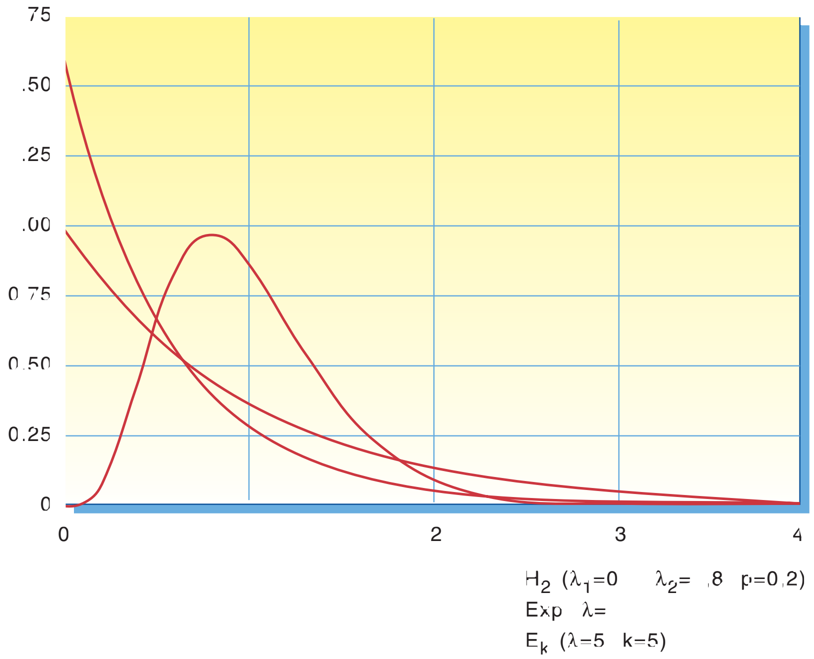


Figure 13 Examples of model distributions for time intervals

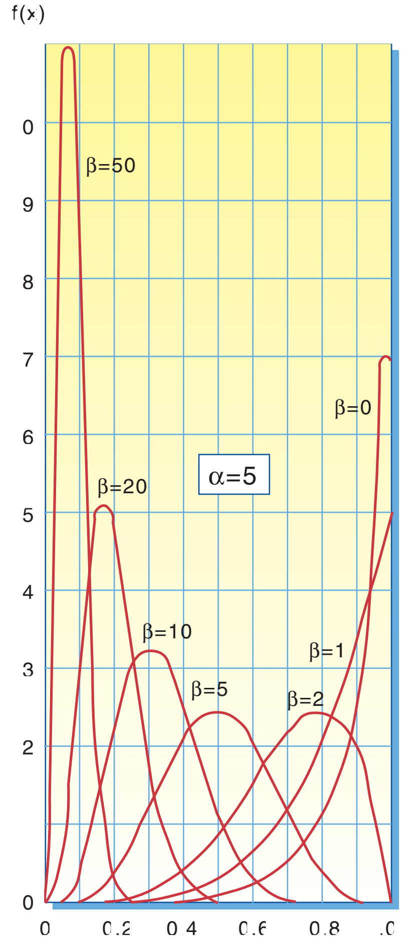


Figure 14 Illustration of β–distribution

The matching between observations and model can be done in several ways. The most common method is to use *moment matching*. For a perfect match in the general case an unlimited number of moments may be required. That is not practicable, and usually only one (mean) or two moments (mean and variance) are applied. In some cases higher moments carry important additional information, in particular the third moment, indicative of the *skewness* of the distribution. It has even been demonstrated that a matching can be improved by omitting, say, third moment and instead use a higher order moment, say fifth or seventh.

The number of moments used in a matching process depends of course on the mathematical distribution in question. The exponential distribution and the Poisson distribution are one-parameter distributions, and matching is limited to one moment. However, more moments may be compared to see how well the choice of distribution fits with the observations. Examples of two-parameter distributions are log-normal, special Erlangian and Weibull among those mentioned above. General Erlangian, hyperexponential and Cox distributions have no upper limit on the number of parameters.

If a random variable X in an experiment comes out with the r sample values X_1, X_2, ..., X_r, then the n^{th} ordinary moment (related to 0) is given by

$$M_n = \frac{1}{r} \cdot \sum_{i=1}^{r} X_i^n \qquad (5)$$

and the n^{th} central moment (related to the mean value $\mu = M_1$) is

$$m_n = \frac{1}{r} \cdot \sum_{i=1}^{r} (X_i - \mu)^n \qquad (6)$$

For an unlimited sample distribution ($r \rightarrow \infty$) the weighting factor per actual X-value, instead of $1/r$, is the probability $p(x)$ for a discrete distribution, and per infinitesimal interval dx a factor $f(x) \cdot dx$ for a continuous distribution with probability density $f(x)$. The corresponding expressions then are

$$M_n = \sum_{x=0}^{\infty} x^n \cdot p(x), \text{ or } M_n = \int_0^{\infty} x^n \cdot f(x)dx \qquad (7)$$

$$m_n = \sum_{x=0}^{\infty} (x-\mu)^n \cdot p(x), \text{ or}$$

$$m_n = \int_0^{\infty} (x-\mu)^n \cdot f(x)dx \qquad (8)$$

Based on the moments we define some useful quantities:

Mean value: note that	$\mu = M_1$, $m_1 = M_1 - \mu = 0$
Variance:	$v = M_2 - M_1^2$
Standard deviation:	$\sigma = \sqrt{v}$
Coefficient of variation:	$c = \sigma/\mu$
Peakedness:	$y = v/\mu$
Form factor:	$\varepsilon = 1 + c^2$
Skewness:	$s = m_3/\sigma^3$
Excess (kurtosis):	$e = m_4/\sigma^4 - 3$

For a more specific study of distributions see Frame 1.

Illustration of the distributions discussed in Frame 2 are given in Figures 13 and 14. Other interesting continuous distribu-

Frame 3

Discrete distributions

As intervals are measured in continuous time, events are counted. The probability of r discrete events can be expressed by the frequency function

$$p(r), \quad r = 0, 1, 2, \ldots$$

and for true probabilities we must have

$$\sum_{i=0}^{\infty} p(i) = 1 \qquad (29)$$

The (cumulative) distribution function is

$$P(r) = \sum_{i=0}^{r} p(i) \qquad (30)$$

The n^{th} statistical moment is defined by

$$M_n = \sum_{i=0}^{\infty} i^n \cdot p(i) \qquad (31)$$

The Z-transform

The Z-transform, also denoted probability generating function, is expressed by

$$f(z) = \sum_{i=0}^{\infty} z^i \cdot p(i); \qquad -1 \le z \le +1 \qquad (32)$$

By differentiating with respect to z we obtain

$$F_n = f^{(n)}(z)|_{z=1} = \sum_{i=0}^{\infty} \binom{i}{n} \cdot n! \cdot p(i) \qquad (33)$$

F_n is denoted the n^{th} factorial moment. F_n can be expressed as a linear function of the ordinary moments of orders 1 to n and vice versa:

$$F_n = g(M_1, M_2, \ldots, M_n) \qquad M_n = \gamma(F_1, F_2, \ldots, F_n)$$

Examples:

$$F_1 = M_1 \qquad M_1 = F_1$$

$$F_2 = M_2 - M_1 \qquad M_2 = F_2 + F_1$$

$$F_3 = M_3 - 3M_2 + 2M_1 \qquad M_3 = F_3 + 3F_2 + F_1$$

$$F_4 = M_4 - 6M_3 + 11M_2 - 6M_1 \qquad M_4 = F_4 + 6F_3 + 7F_2 + F_1$$

In some connections binomial moments are more convenient, expressed by

$$\beta_n = F_n/n! \qquad (34)$$

Frame 4

Useful number distributions

Some of the most interesting number distributions in the teletraffic scenario are the *geometric distribution,* the *binomial (Bernoulli) distribution* and the *Poisson distribution*. Also truncated forms are of particular interest.

In an experiment one of two events may come up at each sample, the two events being A with probability a and B with probability $(1 - a)$. The *geometric distribution* may be interpreted as the probability that the event A comes up i times before the first event B. It has the frequency function

$$p(i) = (1 - a) \cdot a^i \qquad (35)$$

with the generating function

$$f(z) = \frac{1-a}{1-za} \qquad (36)$$

and the n^{th} factorial moment

$$F_n = f^{(n)}(z)|_{z=1} = n! \cdot \left(\frac{a}{1-a}\right)^n \qquad (37)$$

The *binomial distribution* has the same assumption of two events A and B, only that a fixed number of N samples are taken, of which i comes out with A and $N - i$ with B. The frequency function becomes

$$p(i) = \binom{N}{i} \cdot a^i \cdot (1-a)^{N-i} \qquad (38)$$

From the Z-transform is obtained the n^{th} factorial moment

$$F_n = \binom{N}{n} \cdot n! \cdot a^n \qquad (39)$$

The first two ordinary moments are

$$M_1 = N \cdot a$$

$$M_2 = Na(Na - a + 1)$$

The *Poisson distribution* can be obtained from the binomial distribution by letting $N \to \infty$ and $a \to 0$ so that the mean value Na is kept constant $Na = \mu$. Assuming an arrival process with rate λ, we have the mean $\mu = \lambda t$:

$$p(i,t) = e^{-\lambda t} \cdot (\lambda t)^i / i! \qquad (40)$$

$$f(z) = \sum_{i=0}^{\infty} e^{-\lambda t} \cdot (z\lambda t)^i / i! = e^{-\lambda t(1-z)} \qquad (41)$$

The factorial moments of the Poisson distribution thus become extremely simple:

$$F_n = (\lambda t)^n \qquad (42)$$

Interestingly, we have the following results for central moments

$$M_1(= \mu) = m_2 = m_3 = \lambda t,$$

$$m_4 = 3 \cdot (\lambda t)^2 + \lambda t$$

Thus for instance the variance-to-mean ratio (peakedness factor) is $y = m_2/\mu = 1$.

(The property $y = 1$ for the Poisson distribution is a similar distinction as the $c = 1$ for the exponential distribution.) For number distributions $y < 1$ signifies a steep distribution, whereas $y > 1$ signifies a flat distribution. For the Poisson distribution the skewness is $s = 1/\sqrt{\lambda t}$.

Table 2 Survey of some distribution characteristics

	Distribution	Mean value μ	Coeff. of variation c	Peakedness y	Skewness s
Continuous distribution	Exponential	$\frac{1}{\lambda}$	1	$\frac{1}{\lambda}$	2
	Erlang-k (normalised)	$\frac{1}{\lambda}$	$\frac{1}{\sqrt{k}}$	$\frac{1}{k\lambda}$	$\frac{2}{\sqrt{k}}$
	Hyperexponential	$\Sigma\frac{a_i}{\lambda_i}$	>1	$>\frac{1}{\lambda}$	>2
Discrete distributions	Geometric	$\frac{a}{1-a}$	$\frac{1}{a}$	$\frac{1}{1-a}$	$\frac{1+a}{\sqrt{a}}$
	Binomial	$N \cdot a$	$\frac{1-a}{Na}$	$1-a$	$\frac{(1-2a)}{\sqrt{Na(1-a)}}$
	Poisson	λt	$\frac{1}{\sqrt{\lambda t}}$	1	$\frac{1}{\sqrt{\lambda t}}$

tions for traffic applications are normal and log-normal distributions. The former is of special interest for estimation of measurement confidence. Log-normal distributions are relevant for time duration, since the time perception tends to be logarithmic, as is confirmed by observations. Also Cox distributions, a combination of serial and parallel, can be useful for adaptation purposes.

10.1 Distributions of remaining time

When the distribution of time intervals between adjacent events is given, the question arises: What are the distributions of

1) the remaining time t after a given time x, and
2) the remaining time t after a random point in the interval.

For case 1) we obtain

$$f(t+x|x) = \frac{f(t+x)}{1-F(x)} \tag{27}$$

with mean value

$$\mu(x) = \frac{1}{1-F(x)} \cdot \int_{t=0}^{\infty} [1-F(t+x)]dt$$

and for case 2)

$$v(t) = \lambda \cdot (1 - F(t)), \quad 1/\lambda = E\{t\}, \tag{28}$$

with mean value

$\mu = \varepsilon/2\lambda$, where $\varepsilon = c^2 + 1 = M_2/M_1^2$

for the interval distribution $F(t)$.

In the exponential case we have

$$\frac{f(t+x)}{1-F(x)} = \frac{\lambda \cdot e^{-\lambda(t+x)}}{e^{-\lambda x}} = \lambda \cdot e^{-\lambda t}$$

and

$$v(t) = \lambda \cdot (1 - F(t)) = \lambda \cdot e^{-\lambda t},$$

in both cases identical to the interval distribution. Only the exponential distribution has this property.

Paradox: Automobiles pass a point P on the road at completely random instants with mean intervals of $1/\lambda = 10$ minutes. A hiker arrives at point P 8 minutes after a car passed. How long must he expect to wait for the next car? Correct answer: 10 (not 2) minutes. Another hiker arrives at P an arbitrary time. (Neither he nor anybody else knows when the last car passed.) How long must he expect to wait? Correct answer: 10 (not 5) minutes. How long, most likely, since last car? Correct answer: 10 minutes. In this latter case expected time since last car and expected time till next car are both 10 minutes. Mean time between cars is still 10 minutes!

11 The arrival process

It might be expected that constant time intervals and time durations, or possibly uniform distributions, would give the simplest basis for calculations. This is by no means so. The reason for this is that the process in such cases carries with it knowledge of previous events. At any instant there is some dependence on the past. A relatively lucky case is when the dependence only reaches back to the last previous event, which then represents a *renewal point.*

Renewal processes constitute a very interesting class of *point processes*. Point processes are characterised by discrete events occurring on the time axis. A renewal process is one where dependence does not reach behind the last previous event. A renewal process is often characterised by *iid*, indicating that intervals are *independent* and *identically* distributed. Deterministic distributions (constant intervals) and uniform distributions (any interval within a fixed range equally probable) are simple examples of renewal processes.

There is one particular process with the property that all points, irrespective of events, are renewal points: this process follows *the exponential distribution*. The occurrence of an event at any point on the time axis is independent of all previous events. This is a very good model for arrivals from a large number of independent sources. In fact, it is easily shown that the condition of independent arrivals in two arbitrary non-overlapping intervals necessarily leads to an exponential distribution of interarrival intervals.

This independence property is also often termed the *memoryless* property. This implies that if a certain time x has elapsed since the last previous event, the remaining time is still exponentially distributed with the same parameter, and the remaining time from any arbitrary point has the same distribution. The same applies to the time back to the last previous event. It may seem as a paradox that the forward and backward mean time to an event from an arbitrary point are both equal to the mean interval, thus implying twice the mean length for this interval! The intuitive explanation is the better chance of hitting the long intervals rather than the short ones.

The exponential distribution for time intervals between adjacent events implies that the number of events during a fixed length of time is in agreement with the Poisson distribution. The two distributions are said to be equivalent. A simple demonstration of this is seen from the zero event Poisson expression

$$p(0,t) = e^{-\lambda t} \cdot (\lambda t)^i/i! \,\big|_{i=0}$$
$$= e^{-\lambda t} = G(t) = Pr\{T > t\}$$

leading to the exponential survivor function. In other words: The probability of no arrivals during time t is equal to the probability of the time to the next event being greater than t.

The "simple stream"
Khintchine [4] specified the conditions for what he denoted a *simple stream* by three terms: 1) stationarity, 2) absence of aftereffects (independence) and 3) orderliness.

Conditions 1) and 2) suffice to specify the exponential interval between events, whereas condition 3) simply specifies single arrivals. The absence of aftereffects at any point in time is also termed the *Markov* property. An example of an observed distribution with a matching exponential curve is given in Figure 15 [5].

The batch process
It is possible to separate each of the three conditions. Departure only from condition 3) may indicate a *batch* process. It does not influence the other properties.

The non-stationary stream
Condition 1) implies that the stochastic description of the process at time $(t + \tau)$ is identical to that of time t, where τ is an arbitrary interval. Condition 2) indicates a Poisson process. If condition 2) is fulfilled, but not condition 1), we have a Poisson process where the rate $\lambda(t)$ is time dependent. The probability $p(i,\tau)$ of i arrivals during the interval τ is Poissonian with mean $\lambda\tau$, where λ is the mean of $\lambda(t)$ over interval τ. Even though a non-stationary process may have this Poisson property, it does not mean that samples taken from intervals over some period (for instance three minute intervals over an hour) belong to a Poisson distribution. On the contrary, if the rate varies, a peakedness $y = v/\mu > 1$ will always be the case.

The renewal stream
The simple stream is a renewal stream where all points in time are renewal points, as already mentioned above. This is a good model for arrivals from a great number of independent sources. In fact, if no single source is dominant, such a stream will result even if the arrivals from each source are arbitrarily distributed. (The rate from each source approaches zero.)

In a different example, assume a state i defined by i servers being busy in a group of identical servers. We are interested in the distribution of the interval between a departure from i to the first return to i. The first event is, say, an arrival at time t_1 that brings the system to state $i + 1$. If next event is a departure, it happens at time t_2, such that $t_2 - t_1$ is exponential. If on the other hand the next event is an arrival, the system is brought to state $i + 2$, and it may move around through several states $i + j$, $j > 1$ and even several times before it eventually returns to state i. (Similarly the states may be $i - j$ if the first event is a depar-

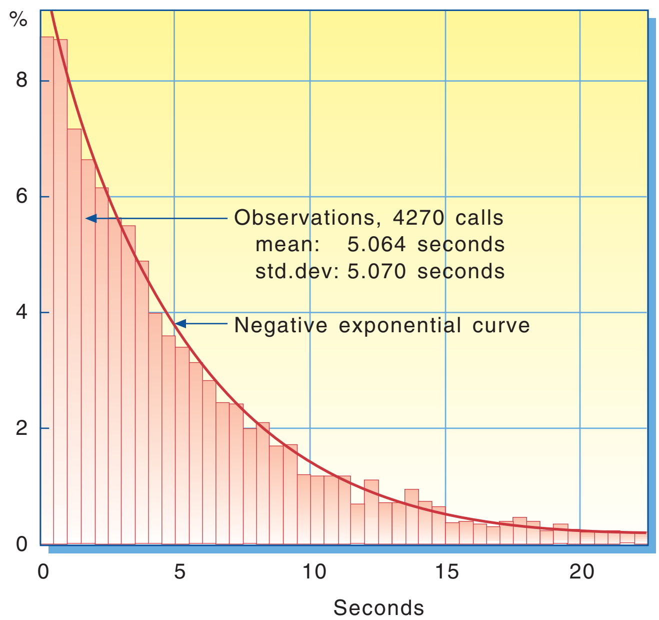


Figure 15 Measured distribution of arrivals in a subscriber group, with matching to an exponential curve

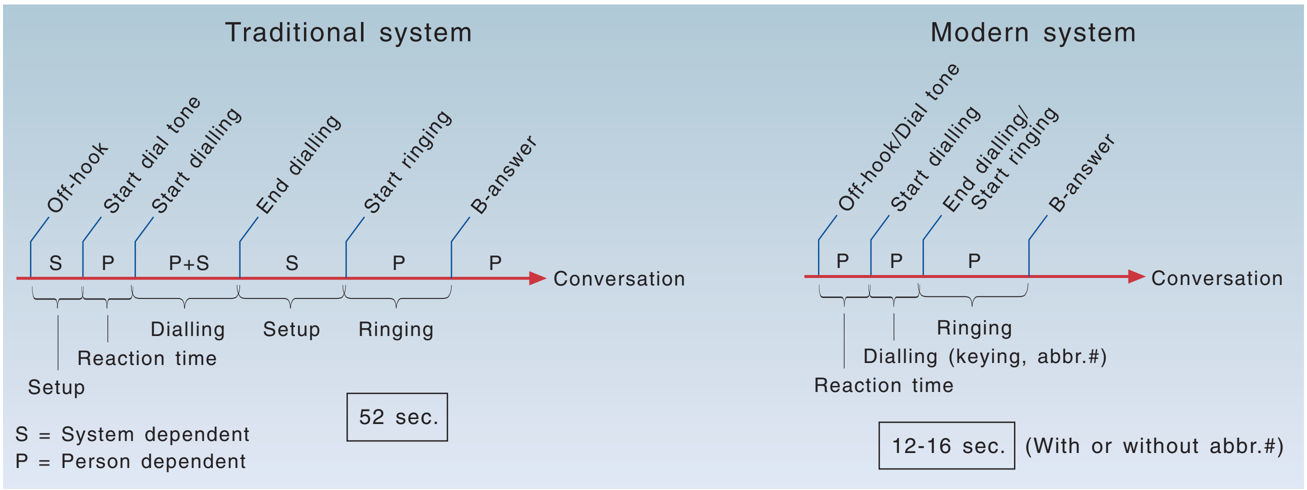


Figure 16 Examples of phased set-up times in a telephone system

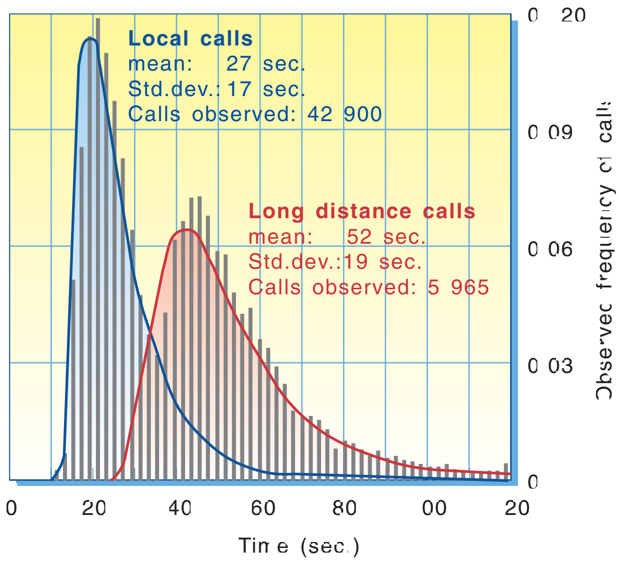


Figure 17 Measured set-up times with matching to log-normal distributions

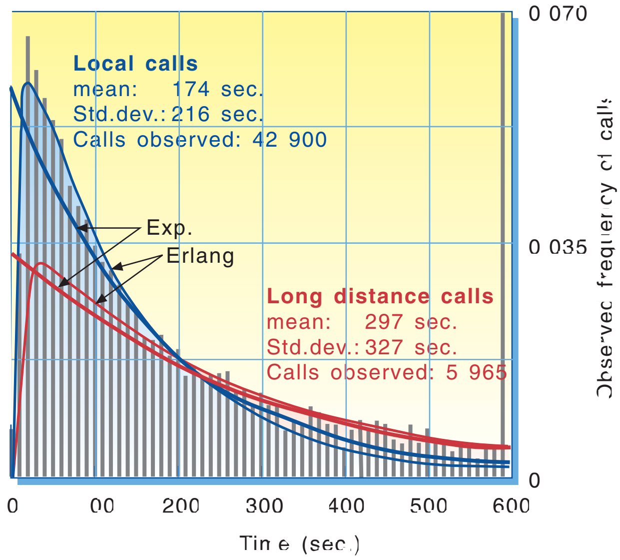


Figure 18 Measured conversation times with matching to exponential and Erlang distributions

ture.) The intervals in case are not from the given exponential distribution. From a stochastic point of view it is not possible to distinguish between the visits in state *i*. Thus each departure from *i* is a renewal point, and the time to the first return to *i* belongs to a non-exponential distribution. The aftereffect within each interval reaches back to this renewal point, but not further back. A renewal process with non-exponential intervals is a case of limited aftereffect.

12 Holding times

The basic arrival and departure processes are shown in Figure 11, with the arrival intervals t_i and the holding times s_i. We shall now have a closer look at holding times.

The time distributions presented earlier for arrival intervals are also applicable to holding times. Thus, exponential distributions and deductions like hyperexponential and Erlangian distributions are primary choices for matching to observations. As we shall see also log-normal distribution has been suggested. The memoryless feature of the exponential distribution implies that however long an interval has lasted, the probability of ending in the next element of time is the same. This does not imply that any lifetime is equally likely!

The special attractiveness of the exponential distribution for analytic purposes very often has made this distribution a default choice.

There is a story about a fellow one night searching for a lost coin under a lamp post. Asked whether he was sure it was the place where he had lost it, the answer was: No, in fact it wasn't, but this is such a good place to look!

Similarly, even if we know that the exponential distribution is not a good approximation, it is often used for convenience. To be true, in many cases it does not matter. In the lost call traffic model with Poisson arrivals the holding time distribution is of no consequence for occupancy distribution and the loss calculation. On the other hand, in most queuing systems it does matter.

Holding times usually consist of several phases, as a minimum a set-up phase and a conversation phase. The set-up phase itself usually also consists of several steps. Two typical examples are given in Figure 16. The first is the set-up procedure in a traditional telephone system, the second is similar for a more modern version.

It should be noted that in the first example the system dependent times *(S)* are very substantial, whereas they are simply negligible in the modern version. The person dependent times *(P)* are perceived as less critical, as they represent a user activity and are not subject to the impatience of waiting. (An exception is ringing time.) The importance of the difference between the two cases may be illustrated by observation results from a university PABX (where the full saving of Figure 15 was still not obtained), showing a time saving on outgoing calls of some 25,000 hours per year for 4,000 extensions.

Examples of set-up times for long distance and local calls with matching to log-normal distributions are shown in Figure 17 [6]. The choice of distribution is a matter of discussion, as a convolution of phases (Erlang-k) might seem more natural. However, the match is fairly good, and the choice might be defended by a probable positive correlation between phases. A similar diagram for *conversation* times is shown in Figure 18. Here a matching to pure exponential and to a two-phase general Erlang distribution are carried out. The tails of >10 minutes are not included in the matching. In fact, as the coefficient of variation, $c > 1$ (c = 1.24 and c = 1.1), a simple Erlangian match would not be possible. The measured times are from observations of long distance calls from a PABX [6].

Two other examples of conversation time distribution for local calls with somewhat larger coefficients of variation (c = 1.34 and c = 1.57) are shown in Figure 19, with a typical shape: With the exponential distribution as a reference the observations show

- fewer very short calls (< 20 seconds)
- an overshoot for calls of short duration (20 –120 seconds)
- an undershoot for calls of 2 – 10 minutes
- a tail of more numerous long-lasting calls.

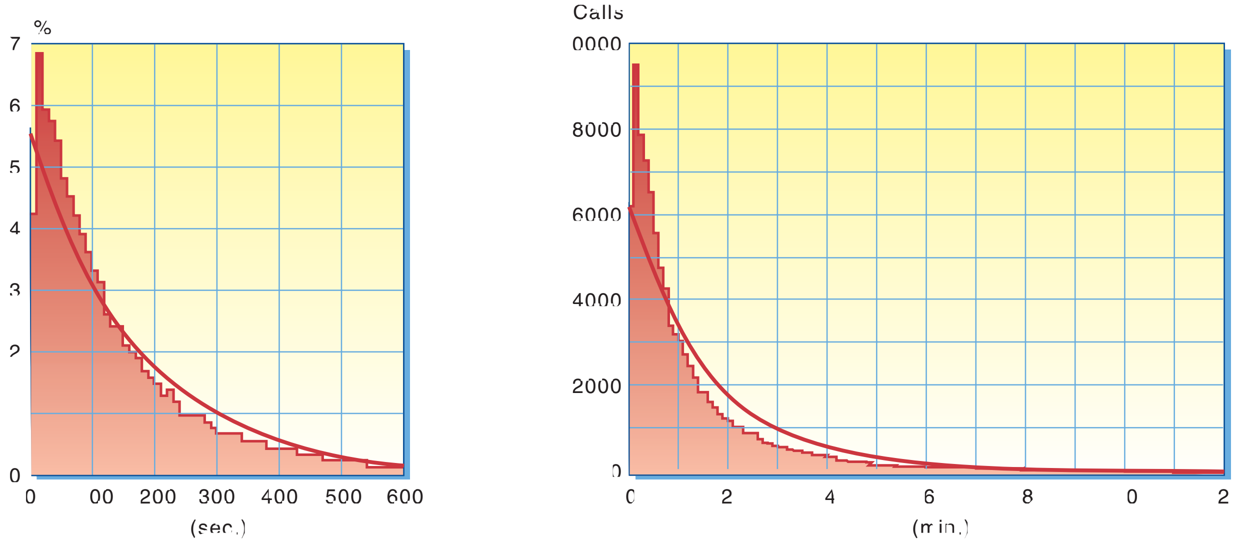


Figure 19 Distribution of conversation times for local calls

This means an accumulation of less than average length and of very long calls.

The diagrams in Figure 19 are taken from [7] and [8]. The latter study shows coefficients of variation from $c = 1.57$ to $c = 3.14$, dependent on time of day, for a mixture of all types of calls, including local to international telephone, and fax, mobile, paging and special services. For normal telephone calls during working hours $c = 1.5 – 1.55$ is typical, with means of $\mu = 3.0$ and $\mu = 4.2$ minutes for local and long distance calls.

A more thorough modelling study of holding time distributions in the telephone network has been carried out by Bolotin [9]. Bolotin's reasoning is that Weber's logarithmic psychophysical law of perception may also apply to perception of conversation time, so that e.g. an increase from 30 to 40 seconds is felt similar to an increase from 3 to 4 minutes. A study of the conversation time for 1000 calls from a single subscriber during 20 working days turns out very similar to the diagrams of Figure 19 for local calls. Bolotin's study contains all calls (local and long distance) and thus has a $c = 1.67$. The match to a log-normal distribution is very good. (A few very short times are excluded.)

For a mixture of different subscriber categories over 15 hour day periods a flatter distribution results. The c-value per day differs between 1.9 and 2.2. In this case a combination of two log-normal distributions is found to give a good match.

Typical of all the results referred to above is the consistent deviation from the exponential distribution. The most characteristic features are an increasing probability of ending during the early phase of a call, changing gradually to the opposite in a later phase. In other words, the longer a call has lasted after an initial phase, the more likely it will continue.

Above is referred to a case where mean holding time after working hours is 1.4 times that of working hours. That is an example of a quite general phenomenon. In Figure 20 is shown an example with a mean holding time of some 25,000 calls changing from 156 seconds to 320 seconds – more than twice – after 17.00 hours. In this case there are probably two effects working together: a transition from work to leisure and a change to a cheaper tariff.

13 Statistical equilibrium

As indicated earlier, a traffic process consists of the two partial processes defined by arrivals and departures, again with reference to Figure 11. We now know a lot more about those two processes. In order to proceed we must look at the traffic carrying system. The simplest case is of course a single server system, which can in fact be very useful if

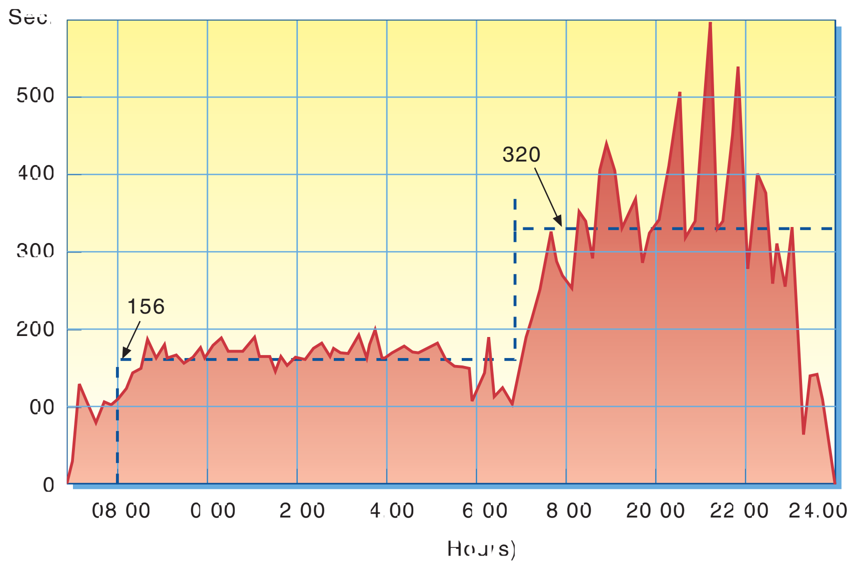


Figure 20 Mean conversation time of local calls

Frame 5

Equilibrium calculations for some typical cases

The Poisson case

The simplest assumption is that the number of sources *(N)* and the number of servers *(n)* are both unlimited, while each source *j* contributes with an infinitesimal call rate (λj), so that the overall call rate is *state independent* $\lambda_i = \lambda = N \cdot \lambda_j$. The departure rate in state *i* is proportional to the number in service, which is of course *i*: $\mu_i = i \cdot \mu = i/s$, where *s* is the mean holding time per call.

The parameters λ and μ are constants, and equation (43) becomes

$$p(i) \cdot \lambda = p(i+1) \cdot \mu \cdot (i+1) \tag{48}$$

leading to

$$\begin{aligned} p(1) &= (\lambda/\mu) \cdot p(0) \\ p(2) &= (\lambda/2\mu) \cdot p(1) = (\lambda/\mu)^2/2! \cdot p(0) \\ &\vdots \\ p(i) &= (\lambda/\mu)^i/i! \cdot p(0) \\ &\vdots \end{aligned}$$

Thus, using equation (44) we get

$$\sum_{i=0}^{\infty} p(i) = p(0) \cdot \sum_{i=0}^{\infty} (\lambda/\mu)^i / i!$$

$$= p(0) \cdot e^{\lambda/\mu} = 1 \Rightarrow p(0) = e^{-\lambda/\mu}$$

and

$$p(i) = e^{-\lambda/\mu} \cdot (\lambda/\mu)^i/i!$$

Using Little's formula: $A = \lambda \cdot s = \lambda/\mu$, we obtain

$$p(i) = e^{-A} \cdot A^i/i! \tag{49}$$

which we recognise as the Poisson distribution with parameter A. The mean value of the distribution is also *A*, as it ought to be. Recalling the interpretation of Little's formula we can state that

$A = \lambda/\mu = \lambda \cdot s$ = average number of customers being served = average traffic intensity.

The truncated Poisson case (the Erlang case)

In practice all server groups are limited, so the pure Poisson case is not realistic. A truncation by limiting the number of servers to *n* leads to a recursive equation set of *n* equations (plus one superfluous) of the same form as that of the unlimited Poisson case above. The normalising condition is different, as

$$\sum_{i=0}^{n} p(i) = 1 \text{ and } \sum_{i=n+1}^{\infty} p(i) = 0$$

The truncated Poisson distribution thus becomes

$$p(i) = \frac{\frac{A^i}{i!}}{\sum_{j=0}^{n} \frac{A^j}{j!}} \qquad 0 \le i \le n$$

$$p(i) = 0, \qquad i > n \tag{50}$$

Any call arriving when $i < n$ will find a free server. Since λ is independent of *i*, the probability of an arbitrary call finding all servers busy is equal to the probability that $i = n$:

$$P(n,A) = E(n,A) = \frac{\frac{A^n}{n!}}{\sum_{j=0}^{n} \frac{A^j}{j!}} \tag{51}$$

This is the *Erlang congestion (loss) formula* for an unlimited number of sources. The formula is not very practical for calculations, so there are numerous tabulations and diagrams available.

In principle one must distinguish between time congestion and call congestion, time congestion being the probability of finding all servers busy at an arbitrary point in time, and call congestion being the probability that an arbitrary call finds all servers busy. In the Erlang case the two probabilities are identical, since all calls are Poisson arrivals, independent of state.

The binomial (Bernoulli) case

We assume a limited number of traffic sources *(N)* and enough servers to cover any need, i.e. $n \ge N$. Furthermore, each *free* traffic source generates λ calls per time unit (λ is here the individual rate, rather than the total rate, and busy sources naturally do not generate any calls). Departure rate per call in progress is unchanged $\mu = 1/s$. The equilibrium equations then take the form

$$p(i) \cdot (N-i) \cdot \lambda = p(i+1) \cdot \mu \cdot (i+1) \qquad 0 \le i \le N \tag{52}$$

By recursion from $i = 0$ upwards we get

$$p(i) = p(0) \cdot \binom{N}{i} \cdot \left(\frac{\lambda}{\mu}\right)^i = p(0) \cdot \binom{N}{i} \cdot b^i \tag{53}$$

$b = \lambda/\mu$ = offered traffic per *free* source

The normalising condition is

$$\sum_{i=0}^{N} p(i) = 1 \text{ and } \sum_{i=N+1}^{\infty} p(i) = 0$$

The $p(i)$-expression above is seen to consist of the common factor $p(0)$ and the i^{th} binomial term of $(1+b)^N$. Thus, by the normalising condition, we obtain

$$p(0) = (1+b)^{-N}$$

and

$$p(i) = \binom{N}{i} \cdot b^i \cdot (1+b)^{-N} = \binom{N}{i} \cdot a^i \cdot (1-a)^{N-i} \tag{54}$$

Here, $a = b/(1+b) = \lambda/(\lambda+\mu)$ = offered traffic per source.

Offered traffic per source takes into account that a busy source does not initiate any calls. Since there are at least as many servers as there are sources, there will be no lost calls. This fact is a clear illustration of the difference between time congestion and call congestion when $N = n$:

Time congestion = $P\{$all servers busy$\} = p(n) = p(N) = a^N$
Call congestion = $P\{$all servers busy when call arrives$\} = 0$
For $N < n$, time congestion = call congestion = 0

The truncated binomial case (the Engset case)

The truncated binomial case is identical to the binomial case, with the only difference being that $N > n$. This condition implies that there are still free sources that can generate calls when all servers are occupied, and there will be call congestion as well as time congestion. The statistical equilibrium equations are the same, whereas the normalising condition is

$$\sum_{i=0}^{n} p(i) = 1 \text{ and } \sum_{i=n+1}^{\infty} p(i) = 0$$

This leads to an incomplete binomial sum and a not quite so simple distribution:

$$p(i) = \frac{\binom{N}{i} \cdot b^i}{\sum_{j=0}^{n} \binom{N}{j} \cdot b^j} \qquad (55)$$

Time congestion can then be expressed by

$$E(N,n,b) = p(i=n) = \frac{\binom{N}{n} \cdot b^n}{\sum_{j=0}^{n} \binom{N}{j} \cdot b^j} \qquad (56)$$

This is the *Engset formula* for a loss system with a limited number of traffic sources. Call congestion B is found, and can be intuitively understood, by deduction from time congestion:

$$B(N,n,b) = E(N-1,n,b) < E(N,n,b) \qquad (57)$$

A modified form similar to that in the binomial case can be expressed by $a = b/(1 + b)$. Offered traffic per source, however, is expressed by

$$a = \frac{b}{1 + b(1 - B)} \qquad (58)$$

and carried traffic per source

$$a_c = \frac{b \cdot (1 - B)}{1 + b(1 - B)} \qquad (59)$$

For the group as a whole offered and carried traffic are $A = N \cdot a$ and $A_c = N \cdot a_c$ respectively.

The Engset formula has one more parameter than Erlang's formula. Tabulation and diagrams will, therefore, be much more extensive.

holding times are very short (<< 1 second) and either the load is low (<< 1 Erlang) or access is organised in a fair and efficient way, e.g. by queuing or polling.

For ordinary telephone traffic mean conversation times are around 2 – 3 minutes. In fact there is a hierarchy of holding times. In a traditional system that is mirrored in an equipment hierarchy on three or four levels:

- conversation carrying equipment (lines, trunks, selectors)
- signal reception and storage equipment (registers, senders. receivers)
- set-up control equipment (selector controls, markers)
- analysers, translators (number analysis and translation, routing).

Very roughly one can assume one order of magnitude difference between each category (say 150, 15, 1.5 and 0.15 seconds). Long conversation times lead to the preference of a multi-server loss system for the first category. For the second category a multi-server queuing system is more appropriate. The third and fourth categories can use single server time sharing system with queuing.

In state-of-the-art equipment only conversation times are the same, whereas the other three categories "collapse" to electronic speed, and thus can use single server time sharing. This means that dimensioning and performance analysis change character. However, they will still be necessary.

In line with these statements we shall proceed to study a fully accessible n-server loss system. With a number N of uncoordinated traffic sources we can never calculate the exact system state at any particular instant, only statistical probabilities.

First of all we must define the possible system states. Since each of the n servers can be in only one of two states, 0 and 1, the number of possible states is 2^n. However, from a service point of view it is not possible to distinguish between the $n!/[i!(n-i)!]$ different states defined by i busy and $n - i$ free servers. Each of those state sets can thus be equated with one state: exactly i busy servers, no matter which. Since $i = 0$ and $i = n$ are included, we have $n + 1$ states altogether. (For a limited access system we may have to distinguish between the individual states within an i-set.) A state transition diagram for the described case is shown in Figure 21.

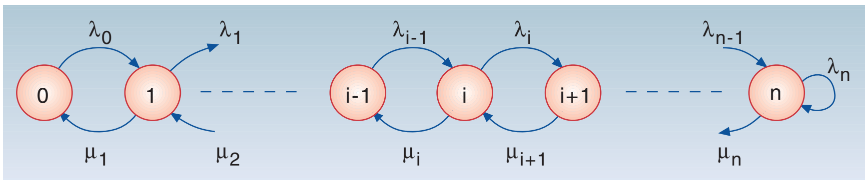


Figure 21 State transition diagram for a fully accessible system

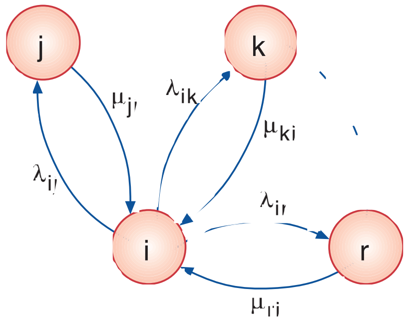


Figure 22 Non-linear state transition diagram

If we assume that in state i we have an arrival rate λ_i and a departure rate μ_i, we obtain a recursive equation, where $p(i)$ is the probability of being in state i:

$$p(i) = p(i-1) \cdot \frac{\lambda_{i-1}}{\mu_i} = \frac{\lambda_0 \cdot \lambda_1 \ldots \lambda_{i-1}}{\mu_1 \cdot \mu_2 \ldots \mu_i}$$
$$\cdot p(0) = \prod_{k=0}^{i-1} \frac{\lambda_k}{\mu_{k+1}} \cdot p(0) \qquad (43)$$

The equation simply says that the rate of going from state i to state $i+1$ must be equal to the rate of going from state $i+1$ to state i, in order to have statistical equilibrium. Otherwise the process would not be stationary. (We use here the Markov property, assuming Poisson arrivals and exponential holding times. Mathematical strictness would require a study based on infinitesimal time elements. That strict approach is omitted here.) The recursion implies an unlimited number of equations when $n \to \infty$. An additional condition, the normalising condition, expresses that the sum of all state probabilities must be equal to one:

$$\sum_{i=0}^{\infty} p(i) = 1 \qquad (44)$$

By combining the two equations we obtain for $p(0)$:

$$1/p(0) = \sum_{i=1}^{\infty} \left\{ \prod_{k=0}^{i-1} \lambda_k/\mu_{k+1} \right\} \qquad (45)$$

to be introduced in the $p(i)$-expression.

Linear Markov chains are assumed, based on the assumption that a state is defined simply by the number of busy servers. Being in state i, an arrival will always change the state to $i+1$, and a departure to state $i-1$. In a non-linear state transition diagram (typically for limited availability systems like gradings), where state i has neighbours j, k, ..., r, equilibrium equations will have the form

$$p(i) \cdot (\lambda_{ij} + \lambda_{ik} + \ldots + \lambda_{ir})$$
$$= p(j) \cdot \mu_{ji} + p(k) \cdot \mu_{ki} + \ldots$$
$$+ p(r) \cdot \mu_{ri} \qquad (46)$$

and normalisation

$$\sum_{\forall i} p(i) = 1 \qquad (47)$$

where $\forall i$ covers all states in the complete state diagram (Figure 22).

In Frame 5 calculations are carried out for four specific cases, using statistical equilibrium, to determine the distributions of the number of customers in the system. The Poisson case assumes no system limitations and hence no congestion. The binomial case leads to time congestion, but no call congestion in the border case of $n = N$, otherwise no congestion. The truncations of Poisson and binomial distributions, characterised by finite $n < N$, both lead to time con-

Table 3 Characteristics of typical traffic distributions

Case	Condition	Distribution	Time congestion *(E)* Call congestion *(B)*
Poisson	$N \to \infty$ $n \to \infty$ $\lambda_i = \lambda$ (total) $\mu_i = i \cdot \mu$	$p(i) = e^{-A} \cdot \frac{A^i}{i!}$ $A = \frac{\lambda}{\mu} = \lambda \cdot s$ = mean value	*E:* None *B:* None
Erlang	$N \to \infty$ n limited $\lambda_i = \lambda$ (total) $\mu_i = i \cdot \mu$	$p(i) = \frac{\frac{A^i}{i!}}{\sum_{j=0}^{n} \frac{A^j}{j!}}$ $A = \frac{\lambda}{\mu} = \lambda \cdot s$	$E(n,A) = p(i = n)$ $B(n,A) = E(n,A)$
Bernoulli	N limited $n \geq N$ $\lambda_i = (N - i) \cdot \lambda$ $\lambda = \lambda$ (free source) $\mu_i = i \cdot \mu$	$p(i) = \binom{N}{i} \cdot a^i \cdot (1-a)^{N-i}$ $b = \frac{\lambda}{\mu}$ $a = \frac{b}{1+b} = \frac{\lambda}{\lambda + \mu}$	$E(N,n,A) =$ $p(i = n) = p(N) = a^N$ (for $n = N$, else $E = 0$) $B = 0$
Engset	N limited $n < N$ $\lambda_i = (N - i) \cdot \lambda$ $\lambda = \lambda$ (free source) $\mu_i = i \cdot \mu$	$p(i) = \binom{N}{i} \cdot \frac{b^i}{\sum_{j=0}^{n} \binom{N}{j} \cdot b^j}$ $b = \frac{\lambda}{\mu}$ $a = \frac{b}{1 + b(1-B)}$	$E(N,n,b) = p(i = n)$ $B(N,n,b) = E(N-1,n,b)$

gestion, as determined from the distributions by $p(i = n)$. Call congestion is given by the ratio of all calls that arrive in state $i = n$. The main cases when call congestion is different from time congestion are

- arrivals are Markovian, but state dependent (binomial, Engset)
- arrivals are non-Markovian.

An illustrating intuitive example of the latter case is when calls come in bursts with long breaks in between. During a burst congestion builds up and many calls will be lost. The congested state, however, will only last a short while, and there will be no congestion until the next burst, thus keeping time congestion lower than call congestion. The opposite is the case if calls come more evenly distributed than exponential, the extreme case being the deterministic distribution, i.e. constant distance between calls. These cases are of course outside our present assumptions.

The four cases in Frame 5 are summarised in Table 3. Some comments:

The Poisson case:

1 With the assumptions of an unlimited number of servers and a limited overall arrival rate, there will never be any calls lost.

2 The memoryless property of the Poisson case for arrivals does *not* apply to the Poisson case for number of customers in the system. Thus, if we look at two short adjacent time intervals, the number of arrivals in the two intervals bear no correlation whatsoever, whereas the number in system of the two intervals are strongly correlated. Still both cases obey the Poisson distribution.

3 The departure process from the unlimited server system with Poisson input is a Poisson process, irrespective of the holding time distribution. (The $M/G/\infty$ system.)

4 The Poisson distribution has in some models been used to estimate loss in a limited server case (n). Molina introduced the concept "lost calls held" (instead of "lost calls cleared"), assuming that a call arriving in a busy state stays an ordinary holding time and possibly in a fictitious manner moves to occupy a released server. The call is still considered lost! The model implies an unlimited Markov chain, so that the loss probability will be

$$P\{loss\} = \sum_{i=n}^{\infty} p(i) = e^{-A} \sum_{i=n}^{\infty} A^i / i! \qquad (60)$$

The Molina model will give a higher loss probability than Erlang.

The Erlang case:

1 $E(n,A) = B(n,A)$ is the famous Erlang loss formula. In the deduction of the formula it is assumed that holding times are exponential. This is not a necessary assumption, as the formula applies to any holding time distribution.

2 If the arrival process is Poisson and the holding times exponential, then the departure process from the Erlang system is also Poisson. This also applies to the corresponding system with a queue. (The $M/M/n$ system, Burke's theorem.)

The Bernoulli case:

The limited number of sources, being less than or equal to the number of servers, guarantees that there will never be any lost calls, even though all servers may be busy. The model applies to cases with few high usage sources with a need for immediate service.

The Engset case:

This case may be considered as an intermediate case between Erlang and Bernoulli. There will be lost calls, with a loss probability less than the time congestion, since $B(N,n,b) = E(N-1,n,b) < E(N,n,b)$. A curious feature is the analytic result that offered traffic is dependent on call congestion. The explanation is the assumption that the call rate is fixed for free sources, and with an increasing loss more calls go back to free state immediately after a call attempt.

Tabulations and diagrams have been worked out for Engset, similar to those of Erlang, however, they are bound to be much more voluminous because of one extra parameter.

In principle the Erlang case is a theoretical limit case for Engset, never to be reached in practice, so that correctly Engset should always be used. For practical reasons that is not feasible. Erlang will always give the higher loss, thus being on the conservative side.

A comparison of the four cases presented in Frame 5 is done in Figure 23, assuming equal offered traffic and number of servers (in the server limited cases).

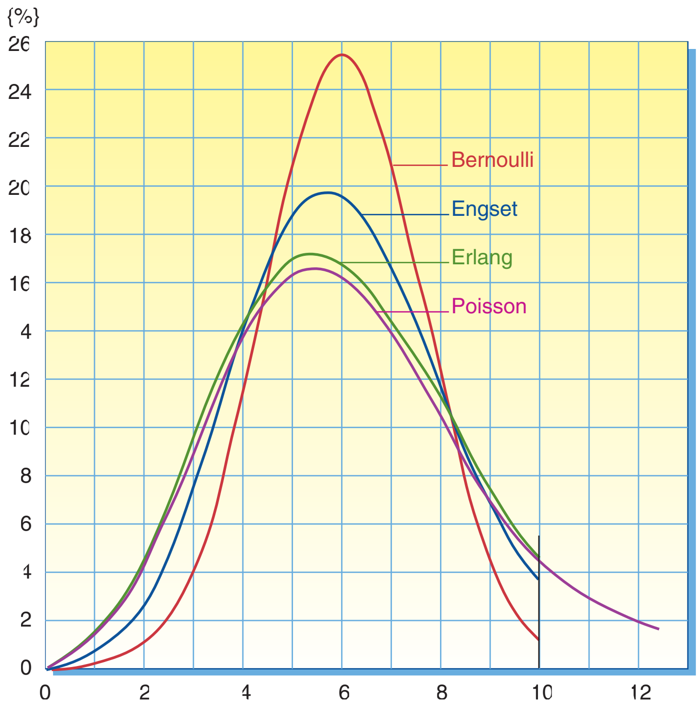


Figure 23 Comparison of different traffic distributions

14 Disturbed and shaped traffic

Up till now we have looked at traffic generation as the occurrence of random events directly from a group of independent free sources, each event leading to an occupation of some server for an independent time interval. Any call finding no free server is lost, and the source returns immediately to the free state. The traffic generated directly by the set of sources is also called *fresh* traffic.

All sources in the group are assumed to have equal direct access to all servers in the server group. This is a *full availability* or *full accessibility* group. (The term *availability* is used in a reliability context as a measure of up-time ratio. In a traffic context availability is usually synonymous with accessibility.) The four models discussed previously are the most feasible ones, but not the only possible.

In a linear Markov chain of $n + 1$ states (a group of n servers) all states have two neighbours, except states 0 and n that have only one. The dwelling time in any state i is exponential with mean $1/(\lambda_i + \mu_i)$, where $\mu_0 = \lambda_n = 0$. (Arrivals in state n get lost and do not influence the system, and there can be no departures in state 0.) When the system changes to state i from $i - 1$ or $i + 1$, it may go direct to $i + 1$ after an exponential interval, or it may first go to $i - 1$ and possibly jump forth and back in states $i, i - 1, i - 2, \ldots$ until it eventually returns to state $i + 1$. It is thus obvious that the interval elapsed between a transition to i ($i > 0$) and a transition to $i + 1$ is not exponential. It is, however, renewal, since it is impossible to distinguish between different instances of state i. There is no memory of the chain of events that occurred before an arrival in state i.

Assume a group of n servers split in a *primary* group P of p servers and a *secondary* group S of $s = n - p$ servers (Figure 24). In a sequential search for a free server an s-server will only be seized if the whole p-group is occupied. There will be two distinct situations in the moment of a new arrival:

1 arrivals and departures in P keep the number of busy servers in the primary group $< p$ (there may be ≥ 0 busy servers in S)

2 all servers in P are busy.

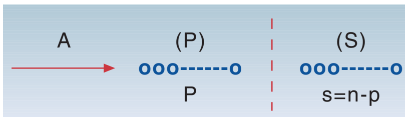


Figure 24 A server group split in a primary group (P) *and a secondary group* (S)

During situation 1) all calls go to P, while in situation 2) all calls go to S. The primary group P will see a Poisson arrival process and will thus be an Erlang system. By the reasoning above, however, the transitions from situation 1) to situation 2), and hence the arrivals to group S, will be a non-Poisson renewal process. In fact, in an Erlang system with sequential search, any subgroup of servers after the first server will see a renewal, non-Poisson, arrival process.

The secondary group S is often termed an *overflow* group. The traffic characteristic of the primary group is that of a truncated Poisson distribution. The peakedness can be shown to be

$$y_p = 1 - A\,[E(p-1,A) - E(p,A)], \quad (61)$$

where $y_p < 1$, since $E(p-1,A) > E(p,A)$

The overflow traffic is the traffic lost in P and offered to S:

$$A_s = A \cdot E(p,A)$$

and the primary traffic is

$$A_p = A - A_s = A[1 - E(p,A)]$$

For A_s the peakedness is $y_s > 1$. (To be discussed later.) A consequence of this is a greater traffic loss in a limited secondary group than what would be the case with Poisson input. The primary group traffic has a *smooth* characteristic as opposed to the overflow that is *peaked.* It is obvious that it is an advantage to handle a smooth traffic, since by correct dimensioning one can obtain a very high utilisation. The more peaked the traffic, the less utilisation.

The main point to be noted is that the original Poisson input is disturbed or shaped by the system. If the calls *carried* in the P-group afterwards go on to another group, the traffic will have a smooth characteristic, and the loss experienced will be less than that of an original traffic offer of the same size.

There are many causes for deviation from the Poisson character, of which some are already mentioned:

- The source group is limited
- Arrivals are correlated or in batches (not independent)
- Arrival rate varies with time, non-stationarity
- Access limitation by grading
- Overflow
- Access limitation by link systems
- Repeated calls caused by feedback
- Intentional shaping, variance reduction.

We shall have a closer look at those conditions, though with quite different emphasis.

14.1 Limited source group

The case has been discussed as Bernoulli and Engset cases. The character is basically Poisson, but the rate changes stepwise at each event.

14.2 Correlated arrivals

There may exist dependencies between traffic sources or between calls from the same source. Examples are e.g.:

- several persons within a group turn passive when having a meeting or other common activity
- in-group communication is substantial (each such call makes two sources busy)
- video scanning creates periodic (correlated) bursts of data.

The last example typically leads to a recurring cell pattern in a broadband transmission system.

14.3 Non-stationary traffic generation

Non-stationarity are essentially of two kinds, with fast variations and slow variations. There is a gradual transition between the two. Characteristic of the fast variation is an instantaneous parameter change, whether arrival rate or holding time. A transition phase with essentially exponential character will result.

An example of instantaneous parameter change is when a stable traffic stream is switched to an empty group. Palm has defined an *equilibrium traffic* that changes exponentially with the holding time as time constant. If thus the offered traffic changes abruptly from zero to $\lambda \cdot s$, then the equilibrium traffic follows

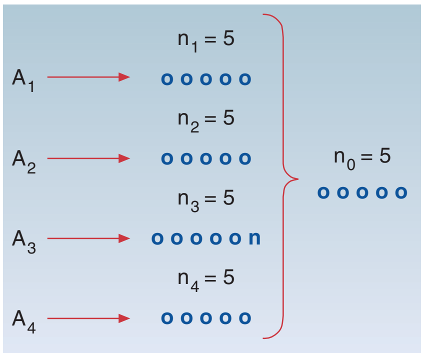


Figure 25 A simple progressive grading with four primary groups and one common (secondary) group

an exponential curve $A_e = \lambda \cdot s \cdot (1 - e^{-t/s})$. In a similar way the equilibrium traffic after a number of consecutive steps can be expressed as a sum of exponential functions. The real traffic will of course be a stochastic variation around the equilibrium traffic. If the server group is limited, an exact calculation of the traffic distribution and loss lies outside the scope of this presentation.

It is of course possible that holding times change with time. Typical is the change at transition from business to leisure and at the time of a tariff change. (See Figure 20.) The change of holding time influences the stationary offered traffic and the time constant as well.

Traffic with slow variations are sometimes termed *quasi-stationary*. The assumption is that the offered traffic varies so slowly that the difference to the equilibrium traffic is negligible. Then the loss over a period can be found with sufficient accuracy by integration based on Erlang's loss formula for each point. In practice one might use a numerical summation based on a step function instead of integration, which would require an unlimited amount of calculation. If the time variation goes both up and down, the exponential lag tends to be compensated by opposite errors.

14.4 Access limitations by grading

Grading is a technique that was developed as a consequence of construction limitations in mechanical selector systems. If a circuit group carrying traffic in a given direction is greater than the number of available outlets for that direction on the selector, then the circuit group must be distributed in some way over the selectors.

Example:
A selector allows $c = 10$ outlet positions for a direction. The total traffic from all selectors of the group in the given direction requires $n = 25$ circuits. A possible grading is obtained by dividing the selector group into four subgroups, each with five circuits per subgroup, occupying $c/2 = 5$ selector positions. The remaining $25 - 20 = 5$ circuits may then occupy the remaining $c - c/2 = 5$ selector positions on all selectors in the four subgroups. The case is shown in Figure 25.

For this grading to be efficient it is assumed sequential hunting first over the dedicated subgroup circuits and then over the common circuits. Each subgroup will thus fully utilise its five allotted circuits before taking any of the common five circuits. In a full availability group a new call over any single selector will always be able to take a free outgoing circuit. In the described grading, however, there may be individual free circuits in one subgroup, while a new call in another subgroup may find the five individual and the five common circuits occupied. Thus calls may be lost even when one or more of the 25 circuits are free.

An important observation is that two states with the same number i of occupied circuits are no longer equivalent, and the simple linear state transition diagrams are no longer adequate. In fact, the simple $n + 1 = 26$-state diagram is changed to one of $(c/2 + 1)^5 = 6^5 = 7776$ states. This is a good illustration of the complications that frequently arise with limited availability. For comparison the maximum number of individual states in a 25-circuit group is $2^{25} = 33{,}554{,}432$.

The grading example is one of a class called *progressive*, characterised by an increasing degree of interconnection in a sequential hunting direction from a fixed start point. With a cycling start point or with random hunting more symmetrical types of interconnection are applied. For analysis, the method of state equations based on statistical equilibrium is of limited value because of the large number of states. Symmetries and equivalencies should be utilised. Several methods have been developed, classified as weighting methods and equivalence methods. Like in other cases when exact analytic methods are not feasible, simulation is an important tool with a double purpose:

- Solving particular problems
- Confirming the validity of approximation methods.

A frequently used analytic method for gradings is Wilkinson's equivalence method, which will be presented under the section on overflow and alternative routing. Grading within switching networks is nowadays of decreasing interest, because of more flexible electronic solutions and multiplexing methods.

14.5 Overflow systems

In Figure 26 is shown three traffic profiles for 5 consecutive working days. The first diagram shows undisturbed traffic on a group of 210 trunks, where a 0.5 % loss line is indicated at 184 Erlang. The measured traffic is at all points well below this line, and there will be virtually no loss. The next diagram shows a 60 trunk group with overflow. The traffic is strongly smoothed, and the group is close to 100 % load during long periods. The third diagram indicates typical strong overflow peaks on a lossless secondary group.

The grading example discussed before consists of four selector groups, each with access to an exclusive group of five circuits and one common group of five circuits (Figure 25). The hunting for a free circuit must be sequential, first over the exclusive group and then over the common group, for the grading to be efficient. There will be a common overflow of calls from the four exclusive groups to the common group. Such overflow is discussed in the beginning of this chapter, where it is pointed out that any overflow after the first server in a group with Poisson input is renewal, non-Poisson. In the example there are four such overflow streams. They are assumed to be mutually independent.

If for a moment we look at a single group with Poisson input, we can easily calculate carried and lost traffic on any part of the server group. If for instance the offered traffic is A, then the carried traffic on the first server is

$$A_c(1) = A - A \cdot E(1, A) = \frac{A}{1 + A}$$

and the overflow

$$A_{ov}(1) = A - A_c(1) = \frac{A^2}{1 + A}$$

The traffic carried on the j servers no. $i + 1$ to $i + j$ is likewise

$$A_j = A \cdot E(i,A) - A \cdot E(i + j,A)$$
$$= A \cdot [E(i,A) - E(i + j,A)]$$

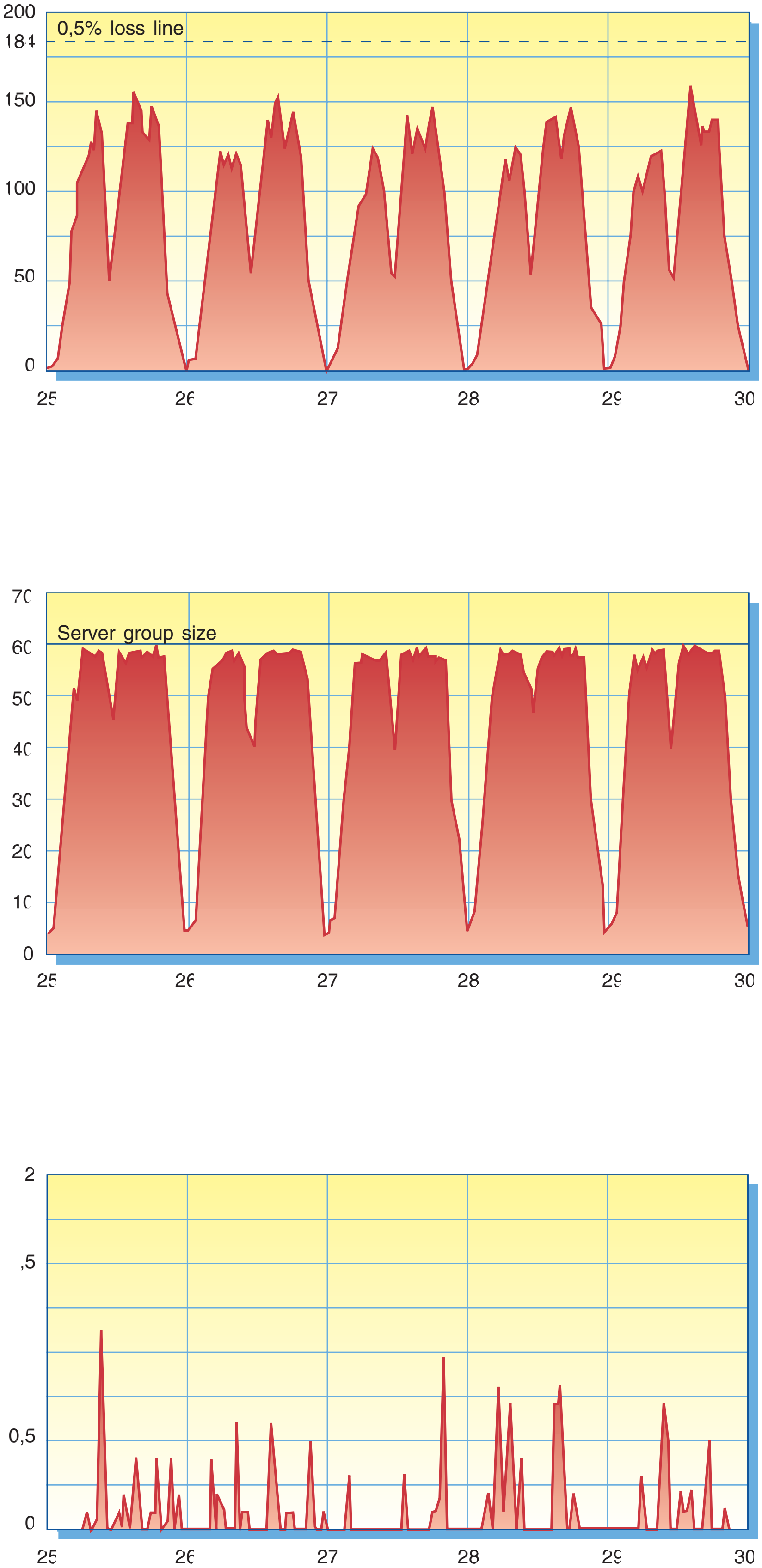


Figure 26 Observed undisturbed (Poisson) traffic, smoothed traffic and overflow

The traffic offered to those *j* servers is thus the overflow from the initial *i* servers:

$A \cdot E\,(i,A) = A'$

If this traffic were fresh Poisson traffic, the carried traffic would be

$A_j' = A' \cdot [1 - E(j,A')]$

Example:

$A = 5$ Erlang, $i = 5$, $j = 1$

$A_j = 5 \cdot [\mathrm{E}(5, 5) - \mathrm{E}\,(5 + 1, 5)]$
$= 5 \cdot (0.284868 - 0.191847)$
$= 0.465$ Erlang

$A' = 5 \cdot \mathrm{E}\,(5, 5) = 5 \cdot 0.284868$
$= 1.42434$

$A_j' = 1.42434 \cdot [1 - \mathrm{E}\,(1, 1.42434)]$
$= 1.42434 \cdot [1 - 0.587]$
$= 0.588$ Erlang

The example illustrates that the traffic carried by a single server is smaller when the offered traffic is an overflow than when it is fresh traffic ($A_j/A_j' = 0.465/0.588 = 0.79$). The offered traffic mean is the same in both cases. As this applies to any server in the sequence, it will also apply to any server group. In other words, overflow traffic suffers a greater loss than fresh traffic does.

This property of overflow traffic can be explained by the greater peakedness (variance-to-mean ratio) of the overflow. This is the opposite property of that of the carried traffic in the primary group, as shown in equation (61). Calculation of the variance is not trivial. The formula is named Riordan’s formula and is remarkably simple in its form:

$$V = M \cdot \frac{1 - M + A}{n + 1 - A + M} \qquad (62)$$

where

A = traffic offered to the primary group (Poisson type)

n = the number of servers in the primary group

M $= A \cdot E\,(n,A)$ = mean value of the overflow.

We note that n and A are the only free variables, and M must be determined from Erlang’s loss formula, so there is no explicit solution where V is expressed by n and A only.

The variance-to-mean ratio = peakedness $= y = V/M = 1 - M + A/(n + 1 - A + M)$ of an overflow stream is an important

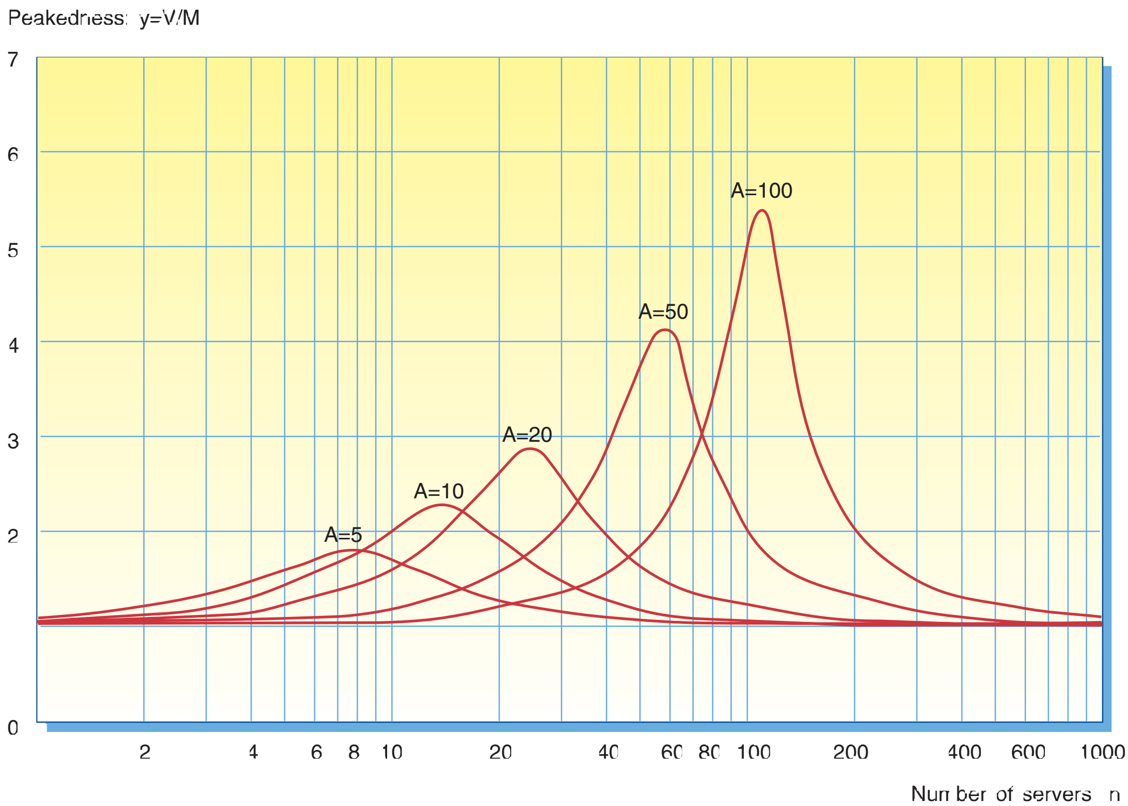


Figure 27 The peakedness y *of overflow as a function of the number* n *of servers for some values of input traffic* A

characteristic, as it compares with the value 1 of a Poisson stream. One might expect y to be monotonously increasing from the first server on. However, this is not so. There is always an initial increase from 1 up to a maximum, after which it decreases towards 1 again. This explains the apparent paradox that the sum of a large number of very small overflow streams may tend to a Poisson stream. A set of peakedness curves is shown in Figure 27. The curves indicate a maximum a bit beyond $n = A$.

The above statement about small overflow streams does not mean that the peakedness of overflows normally approach the 1-limit. As an example take the $A = 20$ Erlang curve of Figure 27. Assume a mean holding time of 120 seconds, which means that the fresh call interval is on average 6 seconds. We get the following set of figures, where the interval τ is average time between overflow calls. Note that even with extreme intervals, there is a fair chance that the next call after an overflow call will also overflow.

n (no. of servers)	τ (interval)	y (peakedness)
0	6 secs	1.0
1	6.3 secs	1.04
10	11 secs	1.61
20	38 secs	2.61
30	12 mins	2.62
40	60 hours	1.75
50	15 years	1.65
60	4000 years	1.49

14.5.1 The ERT (Equivalent Random Traffic) method

If we return to the grading example of Figure 25, we can assume independent input traffics A_1, A_2, A_3 and A_4 to the $g = 4$ primary groups of n_1, n_2, n_3 and n_4 circuits. For each group we can determine mean M_i and variance V_i of the overflow by means of Erlang's and Riordan's formulas. The traffic to the common overflow group of k circuits will according to general statistics be equal to the sum of the means, with a variance (given independence) equal to the sum of the variances:

$$M = \sum_{i=1}^{g} M_i; \tag{63}$$

$$V = \sum_{i=1}^{g} V_i \tag{64}$$

Erlang's formula cannot be used to calculate the loss (overflow) from the overflow group directly, since the input has a peakedness $y > 1$. In statistical terms a Poisson traffic is completely determined by the two parameters λ and μ. In fact, its distribution is uniquely described by the ratio $A = \lambda/\mu$, the first moment, whereby all higher moments are given. Likewise, a single overflow stream is completely determined by the *two* moments mean and variance. All higher moments will then be given. Alternatively, two independent parameters may be given, the obvious choice being the input Poisson traffic A and the size n of the primary group.

It may be noted that even if the substreams in the grading group are renewal and independent, the combined overflow input is not renewal. Thus, this combined

Figure 28 A fictitious equivalent group for generation of a given overflow M,V

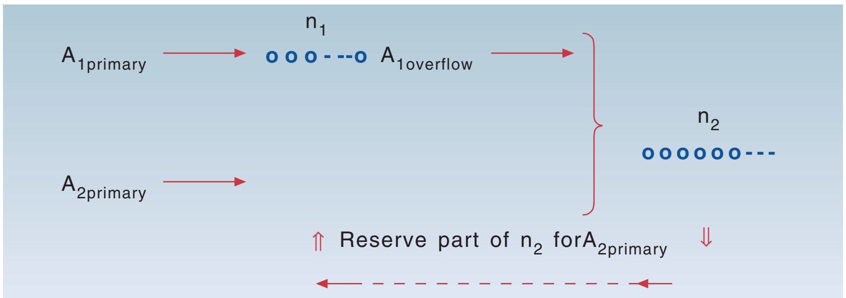


Figure 29 Trunk reservation for primary traffic in order to balance loss

stream is not identical to a single overflow stream with exactly the same mean and variance. However, for practical application the overflow given by the pair (M,V), as calculated above, is assumed to be equivalent to a single overflow stream with the same (M,V) - pair. The principle is shown in Figure 28, where M and V are calculated sum values and the pair (A^*,n^*) represents the equivalent primary group to be determined. Riordan's variance formula gives an explicit expression $A^* = f(n^*)$ or $n^* = g(A^*)$. In principle A^* and n^* are found by insertion in Erlang's formula. This formula, however, is only implicit, and the solution can only be found by an approximation formula, by tables or by numerical calculation.

When A^* and n^* are determined, the overall lost traffic is given by m:

$$m = A^* \cdot E\,(n^* + k,\, A^*)$$

and the loss probability

$$B = m/A = m/\sum_{i=1}^{g} A_i$$

Note that the loss B is the ratio between the calculated lost traffic and the total offered real traffic, m/A, and not related to the fictitious traffic A^* by m/A^*.

The method is called the ERT (Equivalent Random Traffic) method and often referred to as Wilkinson's equivalence method [10]. It can be applied in a stepwise manner for progressive gradings. Another, and at present more interesting area, is that of networks with alternative routing, where primary circuit groups may have overflow to a common secondary group.

14.5.2 The IPP (Interrupted Poisson Process) method

Another method for overflow analysis was introduced by Kuczura [11], and is based on the distinction between the two states $S1$ = [Primary group is busy] and $S2$ = [Primary group not busy]. With the primary input being a Poisson process, the secondary group will see a switching between Poisson input and no input. The method is thus named IPP = Interrupted Poisson Process. The approximation is the assumption that the dwelling times in the two states are exponential, which we know is not true. The process has essentially three parameters, the Poisson parameter λ, and the $S1$ and $S2$ exponential parameters α and β. A three-moment match is based on the mentioned three parameters. An alternative method by Wallstrøm/Reneby is based on five moments in the sense that it uses the first moment and the ratios of 2nd/3rd and 7th/8th binomial moments. [12].

The IPP method is widely used and has been generalised to the MMPP (Markov Modulated Poisson Process) method, where the switching occurs between two non-zero arrival rates.

The methods presented do not assume that all primary groups are equal or near equal. The independence assumption assures the correctness of direct summation of means and variances. Thus, calculation of the overall loss is well founded. In symmetrical cases the overall loss B as calculated above will apply to each of the single primary groups. However, with substantial non-symmetry the single groups may experience quite different loss ratios. This has been studied by several authors, but will not be treated in this paper. Reference is made to [13], this issue.

A particular dimensioning problem of overflow arises in network cases when overflow from a direct route to a final route mixes with a primary traffic on the final, Figure 29. The A_2 traffic may experience a much higher blocking than A_1. The problem can be solved by splitting n_2 and reserving part of it to A_2 only, to obtain a more symmetric grading.

14.6 Access limitations by link systems

Link systems were developed on the basis of crossbar switches and carried on for crosspoint matrices in early electronic switching systems. Like many grading methods, link systems had their motivation in the limitations of practical constructions, modified by the need for improved utilisation of quite expensive switching units.

This can be explained as follows: A traditional selector has one single inlet and, say, 100 to 500 outlets. When used as a group selector, 10 to 25 directions with 10 to 20 circuits per direction would be feasible. All this capacity would still only permit one connection at a time, and a large number of selectors with common access to the outlets (in what is termed a multiple) would be necessary to carry the traffic. A crossbar switch could be organised in a similar way with one inlet and, say, 100 to 200 outlets, and with the same limited utilisation. However, the construction of the crossbar switch (and likewise a crosspoint matrix), permits a utilisation whereby there are, say, 10 inlets and 10 – 20 outlets, with up to 10 simultaneous connections through the switch. This tremendous advantage carries with it the drawback of too few outlets for an efficient grouping. The solution to this is a cascading of switches in 2 – 3 stages. That, of course, brings down the high efficiency in two ways: an increase in the number of switches, that partly "eats" up some of the saving, and introduction of internal blocking, with a corresponding efficiency reduction. Still, there is a considerable net gain.

The selection method through two or more stages is called *conditional selection,* as the single link seizure is conditioned on the whole chain of links being available, and so are seized simultaneously.

The internal blocking mechanism is illustrated in Figure 30. A call generated in stage A searches an outlet in direction x in group C over an available set of links in stage B. If all free links in B are opposite busy circuits in C and vice versa, the call is blocked, even though there are free servers in both stages. The calling individual (traffic source) in A competes with a limited number in the same column for the B-link, and the binomial distribution is normally the natural choice. That means that the number n of sources in an A-column is less than or equal to the number m of links in the B-column. We will assume $n = m$. In stage C, however, all A-stage sources contribute to the load, and the Engset distribution would be feasible. For simplicity Erlang will be preferred, leading to a more conservative dimensioning. Also the assumed independence between stages leads to an overestimation of blocking.

The analysis of a link system is primarily attributed to C. Jacobæus, and a key formula is the Palm-Jacobæus formula

$$H(k) = \sum_{i=k}^{m} p(i) \cdot \frac{\binom{i}{k}}{\binom{m}{k}} \qquad (65)$$

where $H(k) = P\ \{k$ particular out of m circuits are busy, irrespective of the states of the remaining $m - k$ circuits$\}$. In the binomial case (with $n = m$) $H(k) = a^k$.

Internal blocking E between an A-column and a B-column occurs when exactly k circuits in the C-stage and the remaining $m - k$ (and possibly others) in the B-stage are busy. Any value of k, $0 \le k \le m$, may occur:

$$E = \sum_{k=0}^{m} E(k) \cdot H(m-k) \qquad (66)$$

If we use $H(k)$ and introduce the selected distributions above in stages B and C, we obtain the remarkably simple expression for internal blocking (time congestion):

$$E = E\,(m,A) \,/\, E\,(m,A/a) \qquad (67)$$

where A is the offered traffic in the C-column and a is the carried traffic per link in the B-column. Clearly we must have $a < 1$, so that $A/a > A$, and with $m \le n$ the denominator should normally be substantially greater than the numerator, thus keeping E much less than 1, as it ought to be.

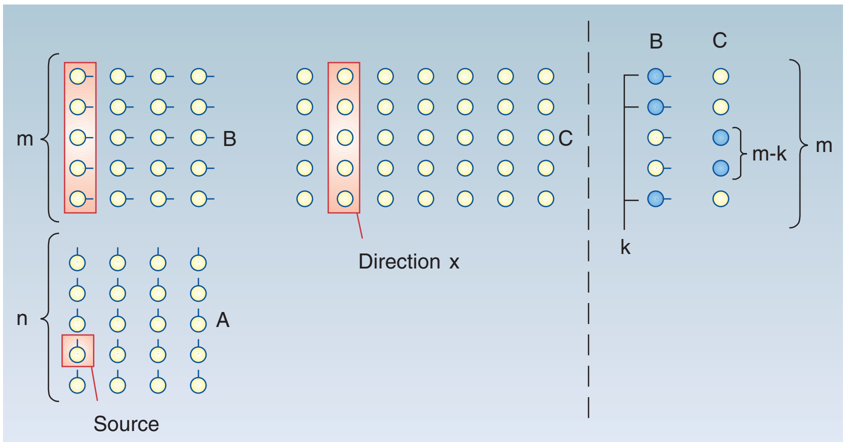


Figure 30 Illustration of internal blocking in a link system

If the number of sources in the A-stage is less than the number of accessible links in the B-stage ($n < m$), the formula is simply modified to

$$E = E\,(m,A) \,/\, E\,(n,A/a)$$

14. 7 Traffic shaping

While the previous effects of traffic disturbance or shaping are unintentional results of source group or system properties, traffic smoothing may be used intentionally to obtain better efficiency and/or better service quality. An unintentional smoothing happens implicitly by means of system limitations that lead to call loss. Loss occurs around traffic peaks to the effect that peaks are cut off. The resulting smoothing leads to better utilisation of following stages.

The most obvious smoothing mechanism is queuing. The simplest way of queuing happens at the traffic source, where the traffic demands may be lined up to give near to 100 % utilisation even on a single server. For comparison, with random calls and no queuing a requirement of 1 % loss will only give 1 % utilisation, and 90 % utilisation will give 90 % loss. If random calls arrive at a queue, the server utilisation can be increased arbitrarily, but only at the expense of increasing waiting time.

Priorities can be used to even out the system load. An early example was introduced in the No. 1Ess computer controlled switching system, where detection of new calls were delayed by low priority in order to avoid overload while calls already in the system were being processed.

In state-of-the-art systems like ATM-based B-ISDN various traffic shaping mechanisms are employed. There are several control levels. Traffic demand may be held back by connection admission control with the objectives of obtaining the intended fairness, quality of service and utilisation. Further shaping is carried out within established connections by means of service related priorities, queuing and discarding of cells. A major distinction has to be done between constant bit rate (CBR) and variable bit rate (VBR) services. Critical CBR services are immediate dialogue services like speech conversation. One-way entertainment services like video and speech transmission are not so delay-critical, whereas delay variations have to be contained by buffering. Less delay-critical are most data services, even interactive services. These often have a bursty character, and there is much to be gained by smoothing through buffering.

A trade-off between delay and cell loss is common, as a CBR service may accept high cell loss and only small delay, whereas the opposite may apply to a VBR service.

14.8 Repeated calls

The traffic shaping effects as discussed so far are due to random fluctuations in combination with system properties and deliberate control functions. A particular type of effects stems from system feedback to users.

One feedback level is that which reduces a traffic *intent* to a lower traffic *demand,* because of low service quality expectations. An abrupt upward change in service quality will demonstrate an upward swing of demand, closer to the intent (which can never be measured).

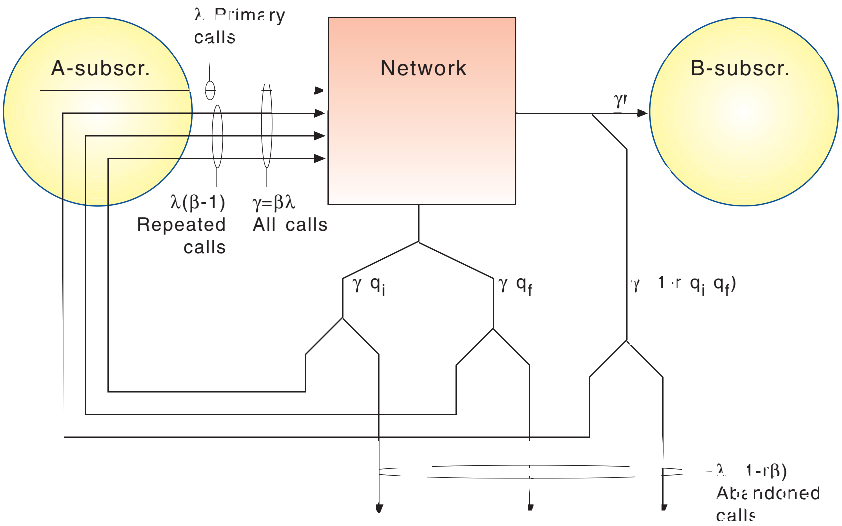


Figure 31 Aggregated model for analysis of repeated calls

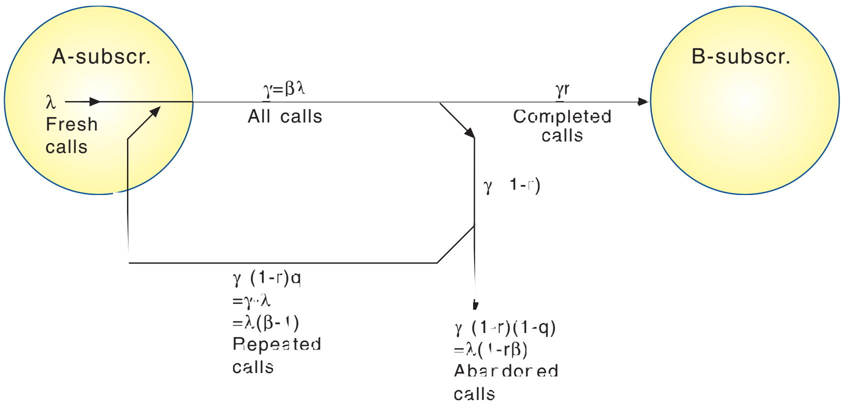


Figure 32 Simplified repeated call model

A feedback effect that can actually be measured is that of repeated calls. The traffic terms listed in Chapter 1 includes several terms related to repeated calls. A call intent may or may not lead to a call demand, resulting in a call string of length n (n = 1,2, ...) that consists of a *first attempt* and $n - 1$ *repeated attempts*. A call attempt may be

- blocked: rejected by the network
- abandoned: aborted by the calling user
- successful (fully routed): information of the called user state is received
- completed (effective): receives an answer signal.

A *successful call attempt* is thus one that has reached the called user, irrespective of whether the user is available for conversation or not. A successful, but uncompleted call attempt is usually one that meets a “busy” or “no answer” state. A *successful call* is a call that has reached the wanted number and allows the conversation to proceed. *Completion ratio* can be referred to some point in the network, and is the ratio of the number of completed call attempts to the total number of call attempts at that particular point.

There is in the literature quite a lot of repeated calls studies. Modelling has been based on purely mathematical considerations, or they have been supported by observation results. Most telephone users are able to distinguish between failure causes when attempting to make a call. An attempted call may result in

- deliberate abandonment
- faulty operation by caller (wrong legal number is not considered a fault)
- network technical failure
- network blocking (congestion)
- busy called user
- no answer
- completion.

The user reaction will depend on his own perception of the situation. He may assume that a technical failure or network blocking will not affect a new attempt, and thus make an immediate retry. A busy user will probably stay busy a few minutes, and it is better to wait a bit. No answer indicates absence, and one might rather wait for half an hour. The worst case is radio or TV announced competition or voting. With abbreviated dialling, repeat key or automatic diallers complete network congestion may result. (I have personally – out of professional interest(!) – made up to a hundred calls in quick sequence with repeat key, with no success.)

A repeated call model is shown in Figure 31. The network normally consists of several stages, and each stage may be considered separately. Here it is considered in aggregated form, and the model is limited to three elements: 1) calling users, 2) network and 3) called users.

The aggregated model can be simplified to the completion/non-completion model in Figure 32, where overall parameters are identified:

λ = primary attempt rate = call string rate

r = efficiency rate = completion rate for all calls

q = probability of repeating a failed call

γ = total attempt rate

β = repetition rate = ratio of all attempts to first attempts

H = persistence.

The parameters λ, r and q are primary parameters, whereas β, γ and H are calculated:

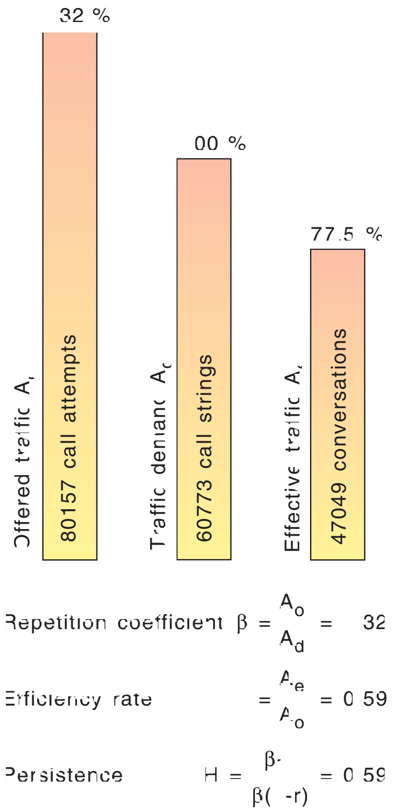


Figure 33 Observation results for repeated calls

$$\gamma = \lambda/[1 - q(1 - r)]$$

$$\beta = \gamma/\lambda = 1/[1 - q(1 - r)] \qquad (68)$$

$$H = (\gamma - \lambda)/\gamma(1 - r) = (\beta - 1)/\beta(1 - r)$$

From the last expression is obtained

$$\beta = 1/[1 - (1 - r)H] \qquad (69)$$

This formula can be interpreted as the result of system properties (including the state of called user) expressed by *completion rate r* and the feedback response of the user expressed by *persistence H*.

An example of observations carried out on a 700 extension PABX is shown in Figure 33.

Observation results [14] indicate that failure (non-completion) rate as well as persistence increases with rank number from first attempt upward. This should not be interpreted as an increasing failure rate and an increasing user persistence within each string. It means that the hard-to-reach destinations and the important call intents among different strings remain, whereas those other demands succeed or are abandoned earlier. This effect may be termed a *selection effect*. The selection effect on completion rate can be calculated by assuming the called destinations to be distributed according to an unavailability between 0 and 1. It can be shown that if the call rate towards a destination is proportional to the unavailability, (which is natural if the cause is a busy state), the failure rate of the i^{th} call in a string is

$f_i = M_{i+1}/M_i$; $M_i = i^{\text{th}}$ ordinary moment

A feasible distribution model is the Beta distribution, as presented in Frame 2.

$$f(x) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\cdot\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

With parameters α and β we obtain the expression

$$f_{i1} = \frac{i+\alpha}{i+\alpha+\beta} \qquad (70)$$

If, however, the call rate towards a destination is state independent, the failure rate of the i^{th} call in a string is

$$f_{i2} = \frac{M_i}{M_{i-1}} = \frac{i+\alpha-1}{i+\alpha+\beta-1} \qquad (71)$$

Similarly, there may be some distribution of the call demands according to urgency. Again, using a Beta distribution model, the persistence can be expressed by

$$H_i = \frac{M_i}{M_{i-1}} = \frac{i+\alpha-1}{i+\alpha+\beta-1} \qquad (72)$$

The mean value of the Beta distribution is $M_1 = \alpha/(\alpha + \beta)$, and the distribution tends to the deterministic distribution when $\alpha \to \infty$, $\beta \to \infty$, so that $\alpha/\beta = M_1/(1 - M_1)$. In such case is seen that $f_i = f =$ constant and $H_i = H =$ constant, independent of i.

Observations indicate that apart from the increasing failure rate with increasing i there is a distinct jump from $i = 1$ to $i = 2$. The reason for this is the state dependence from a failed call to the ensuing repetitions. The explanation is simple: Assume an alternation between two destination states, a failure state lasting an average time $1/\gamma$ and an inter-failure state lasting $1/\delta$. (Exponential distributions are assumed, but Erlangian or two-term hyperexponential also give feasible solutions.) The failure rate of a first attempt, arriving at random is

$$f_1 = \frac{\delta}{\gamma+\delta} \qquad (73)$$

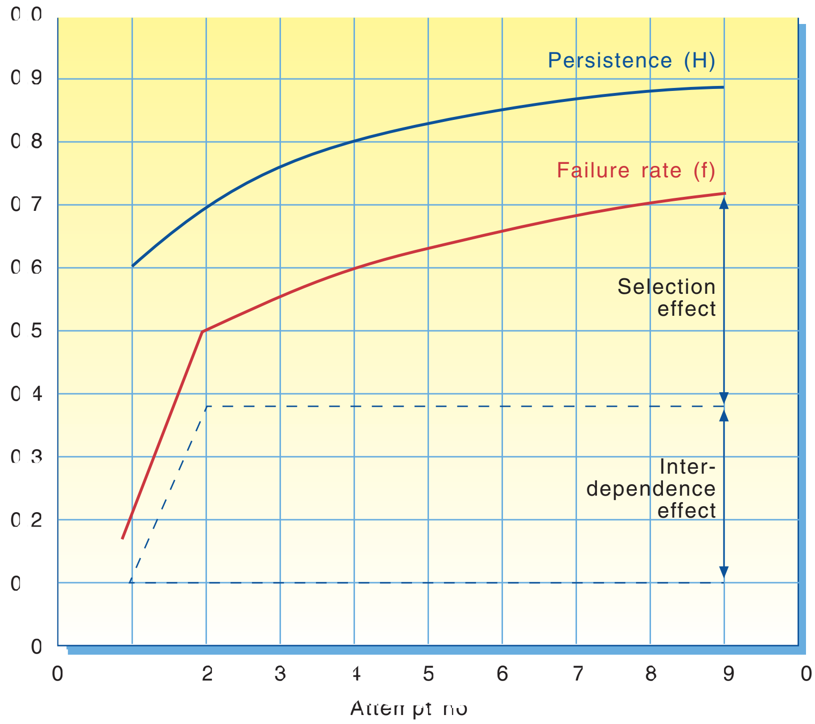


Figure 34 Illustration of interdependence and selection effects for repeated calls

Frame 6

Basic queuing model

From the model given in Figure 12 and the corresponding state transition diagram in Figure 36 we obtain for the two regions:

$$p(i) = (A/i) \cdot p(i-1) = (A^i/i!) \cdot p(0), \ \ 0 \le i \le n$$

and

$$p(i) = (A/n) \cdot p(i-1) = (A^n/n!) \cdot (A/n)^{i-n} \cdot p(0), \ \ n \le i \le \infty \quad (76)$$

with the common expression

$$p(n) = (A^n/n!) \cdot p(0)$$

The normalising condition requires

$$\sum_{i=0}^{\infty} p(i) = 1:$$

$$\frac{1}{p(0)} = \sum_{i=0}^{n-1} \frac{A^i}{i!} + \frac{A^n}{n!} \cdot \sum_{i=n}^{\infty} \left(\frac{A}{n}\right)^{i-n}$$

$$= \sum_{i=0}^{n-1} \frac{A^i}{i!} + \frac{A^n}{n!} \cdot \frac{n}{n-A}$$

since the second sum is a geometric series. This sum is only defined for $A < n$. If $A \ge n$, the geometric series has an unlimited sum.

A random call will be put in the queue whenever the instantaneous state is $i \ge n$. The sum probability of those states are represented by the second term of the above expression multiplied by $p(0)$, so that the waiting probability for a random call is

$$E2(n,A) = p(i \ge n) = p(0) \cdot \frac{n}{n-A} \cdot \frac{A^n}{n!}$$

$$= \frac{\frac{n}{n-A} \cdot \frac{A^n}{n!}}{\sum_{i=0}^{n-1} \frac{A^i}{i!} + \frac{n}{n-A} \cdot \frac{A^n}{n!}} \quad (77)$$

This is Erlang's waiting formula. It turns out that there is a simple relationship between the loss formula and the waiting formula. If the loss and waiting probabilities are termed $E_1 = E_1(n,A)$ and $E_2 = E_2(n,A)$ respectively, then

$$\frac{1}{E_2} = \frac{1 - \frac{A}{n} \cdot (1 - E_1)}{E_1}$$

Thus, as tables and other representations of E_1 are available, E_2 is simply determined by inserting the E_1 value in this relation. The relation also demonstrates that always

$$E_2 > E_1$$

While the simple loss system is solved when the loss probability E_1 is determined, the waiting probability E_2 is only a partial solution, as waiting time and queue length may be even more important. Also the *distributions* of waiting time and queue length are interesting characteristics.

It should be noted that the probability of a random call having to wait is

$$E_2 = p(i \ge n) = \sum_{i=n}^{\infty} p(i) \quad (78)$$

while the probability of one or more customers waiting in the queue is

$$p(i > n) = \sum_{i=n+1}^{\infty} p(i) = E_2 \cdot A/n = p(n) \cdot A/(n-A) \quad (79)$$

Queue length distribution

The distribution of the number of customers in the system is given by the above expressions of $p(i)$, $0 \le i \le \infty$ (Eq. 76). Since the number in queue is 0 when $i \le n$ and $j = i - n$ when $i \ge n$, the mean queue length is calculated by

$$L = \sum_{i=0}^{n} 0 \cdot p(i) + \sum_{i=n+1}^{\infty} (i-n) \cdot \left(\frac{A}{N}\right)^{i-n} \cdot p(n) = E2 \cdot \frac{A}{n-A} \quad (80)$$

The conditional distribution of queue length $j = i - n$, given that there is a queue, is

$$p_q(j) = p(i)/p(i > n) = (A/n)^j \cdot (n-A)/A, \ \ j = i - n > 0$$

The mean queue length, when there is a queue $(i > n)$, thus becomes

$$L_{q1} = \sum_{j=1}^{\infty} j \cdot p_q(j) = \frac{n-A}{n} \cdot \sum_{j=1}^{\infty} j \cdot \left(\frac{A}{N}\right)^{j-1} = \frac{n}{n-A} \quad (81)$$

Averaged over all states $i \ge n$ the length will be

$$L_{q2} = L_{q1} \cdot \frac{A}{n} = \frac{A}{n-A} \quad (82)$$

Waiting time distribution

The distribution of the queue length, given that all servers are busy (including the zero queue) is given by

$$p(i = n + j | i \ge n) = \frac{p(n+j)}{p(i \ge n)} = \left(\frac{A}{n}\right)^j \cdot \frac{n-A}{n}$$

This is seen to be the geometric distribution $p(j) = (1 - a) \cdot a^j$, where $a = A/n$.

A customer arriving when there are j in queue will have to wait $(j + 1)$ exponentially distributed intervals of mean length $1/n\mu$, where $1/\mu = s$ = service time. This gives an Erlang-$(j$+1) distribution, which must be weighted with the j-term of a geometric distribution. Taking the density function we obtain

$$f(t | i \ge n) = \underbrace{\sum_{j=0}^{\infty} (1-a) \cdot a^j}_{\text{Geo. distr.}} \cdot \underbrace{e^{-n\mu t} \cdot \frac{n\mu (n\mu t)^j}{j!}}_{\text{Erlang-}(j+1)\text{ distr.}}$$

$$= \underbrace{\mu (n-A) \cdot e^{-\mu (n-A) \cdot t}}_{\text{Exponential distr.}}$$

The corresponding distribution function is

$$F(t|i \geq n) = 1 - e^{-\mu(n-A) \cdot t} \quad (83)$$

The unconditional distribution function, taken over all states $0 \leq i \leq \infty$, is

$$F(t) = 1 - E_2 \cdot e^{-\mu(n-A) \cdot t} \quad (84)$$

It should be noted that $F(0) = 1 - E_2$. That indicates the probability of zero waiting time, being equal to $p(i < n)$. Thus there is an infinite probability density in $t = 0$. It may look like a lucky – and remarkable – coincidence, that the waiting time distribution, as given by an infinite sum of geometrically weighted Erlang-distributions turn out to give an exponential distribution.

The mean waiting time in queue for those waiting can be taken directly from the distribution function

$$w = 1/\mu(n - A) = s/(n - A) \quad (85)$$

Similarly, averaged on all customers, waiting time will be

$$W = E_2 \cdot w = E_2 \cdot s/(n - A) \quad (86)$$

Alternatively, waiting times can be found by using Little's formula. (Note that the expression of L_{q2}, not L_{q1}, as defined above, must be used):

$$w = \frac{L_{q2}}{\lambda} = \frac{A}{(n-A) \cdot \lambda} = \frac{s}{n-A}$$

$$W = \frac{L}{\lambda} = \frac{E_2 \cdot A}{(n-A) \cdot \lambda} = E_2 \cdot \frac{s}{n-A}$$

The next attempt is not random, but conditioned on the previous one encountering a failure state. Given a calling rate of λ for repetitions it can be shown by convolution that the following failure rates will be given by

$$f_{i>1} = \frac{\lambda_i + \delta_i}{\lambda_i + \delta_i + \gamma_i} \quad (74)$$

This is the *interdependence effect,* causing a jump from first to second attempt. The selection and interdependence effects are superimposed in the failure rate, whereas only selection effect applies to the persistence. The indexing of δ and γ implies the selection effect, while the indexing of λ is just in case there is a change of repetition interval with rank number. A calculated example is given in Figure 34.

Observations show that when the causes "subscriber busy" and "no answer" are compared, "busy" has greater persistence and shorter repetition intervals. Both have a distinct jump in failure rate, but not in persistence, from first to second call. The selection effect of failure probability seems to be stronger for "no answer" than for "busy".

15 Waiting systems and queues

Up till now we have made only sporadic mention of waiting times and queues. In real life those are too well known concepts. Obviously, there are two parts in a service situation, the part offering service and the part seeking service. Both parts want to optimise their situation, that is to maximise benefit and minimise cost. Particularly on the user (service seeking) side convenience and inconvenience come into the calculation beside the purely economic considerations. A simple question is: What is most inconvenient, to be waiting or to be thrown out and forced to make a new attempt? The answer depends on the waiting time and the probability of being thrown out.

There is also the issue of predictability and fairness. In an open situation like that of a check-in counter most people prefer to wait in line instead of being pushed aside at random, only with the hope of being lucky after few attempts at the cost of somebody else. A similar case in telecommunication is that of calling a taxi station. To stay in a queue and get current information of one's position is deemed much better than to be blocked in a long series of attempts.

In telephone practice blocking is used on conversation time related parts like lines, trunks, junctors and switches, whereas waiting is used on common control parts like signal senders and receivers, registers, translators, state testing and connection control equipment, etc. Blocking is preferred because of the cost of tying up expensive equipment and keeping the user waiting. (Waiting time is proportional to holding time.) This requires a low blocking probability, less than a few percent. The high probability of non-completion caused by "busy subscriber" or "no answer" (10 – 70 %!) is a significant problem. It is alleviated by transfer and recall services, voice mail, etc.

Another aspect of queuing is that of service integrity. For real time communication the admissible delay and delay variation is very limited, lest the signal integrity suffers, which severely affects the use of queuing. On a different time scale interactive communication also permits only limited delay. However, it is not a matter of signal deterioration, but rather the introduction of undue waiting for the user.

15.1 Queuing analysis modelling

The loss systems treated up till now are sensitive to the arrival distribution, and we have mostly assumed Poisson arrivals. Traffic shaping, in particular overflow, changes the arrival distribution, with significant consequences for the loss behaviour. With Poisson arrivals a full availability loss system is insensitive to the holding time (service) distribution. Queuing systems, on the other hand, are in general sensitive to arrival distribution and holding time distribution as well. It should be distinguished, therefore, between various cases for both elementary processes.

Kendall introduced a notation of the basic form A/B/C, where *A* indicates the arrival distribution, *B* the service distribution and *C* the number of servers. If nothing else is said, an unlimited queue with no dropouts is assumed. The simplest model for analysis is the M/M/1-system, with Markov input and service and a single server. The notation G/G/n indicates that any distributions are applicable. Since dependence may occur, it

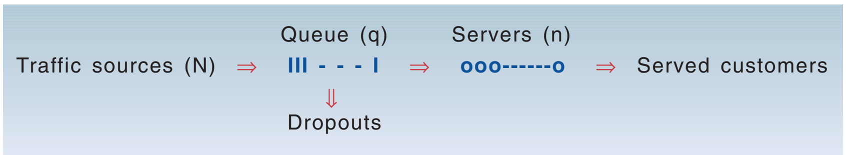


Figure 35 Simple queuing model

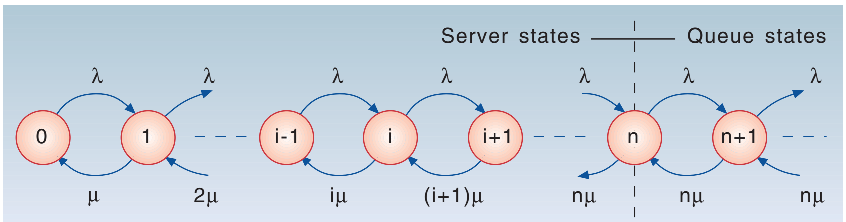


Figure 36 State transition diagram for waiting system with unlimited queue

Table 4 Characteristics of the M/M/n queue

State probabilities	$\frac{1}{p(0)} = \sum_{i=0}^{n-1} \frac{A^i}{i!} + \frac{An}{n!} \cdot \frac{n}{n-A}$ $0 \le i \le n: p(i) = p(0) \cdot \frac{A^i}{i!}$ $n \le i \le \infty: p(i) = p(0) \cdot \frac{A^n}{n!} \cdot \left(\frac{A}{n}\right)^{i-n}$ $= p(n) \cdot \left(\frac{A}{n}\right)^{i-n}$
Probability of waiting	$E_2(n, A) = p(i \ge n) = p(0) \cdot \frac{A^n}{n!} \cdot \frac{n}{n-A}$
Queue length (all-state mean)	$L = E_2 \cdot \frac{A}{n-A}$
Queue length (mean of queue states)	$L_q 1 = \frac{n}{n-A}$
Queue length (mean of all n-busy states)	$L_{q2} = \frac{A}{n-A}$
Waiting time distribution (all waiting calls)	$F(t \mid i \ge n) = 1 - e^{-\mu(n-A) \cdot t}$
Waiting time distribution (all calls)	$F(t) = 1 - E_2 \cdot e^{-\mu(n-A) \cdot t}$
Mean waiting time (all waiting calls)	$w = \frac{s}{n-A}$
Mean waiting time (all calls)	$W = E_2 \cdot \frac{s}{n-A}$

may be appropriate to point out independence by adding I: GI.

The most common indications are:

M ⇒ Markov (Poisson, exponential)
M^b ⇒ Batch-Poisson
D ⇒ Deterministic
E_k ⇒ Erlang-k
H_k ⇒ Hyperexponential-k
Geo ⇒ Geometric.

The Kendall notation may be expanded by additional terms indicating queue length, number of sources, queue discipline, etc. As there is no standard method, expansions are better defined in context.

A simple queuing model is given in Figure 35.

Statistical equilibrium for unlimited M/M/n queue

The common assumption is that of the Erlang case for blocking systems, with the addition of a queue with an unlimited number of positions and with no dropouts. That means an unlimited number of traffic sources with an aggregated call rate λ and a service time $s = 1/\mu$, assuming Poisson input and exponential holding times.

The state transition diagram of Figure 36 is the same as that of the blocking case (Figure 21), with an unlimited addition of queuing states. As no dropouts are assumed, all queue states have the same departure rate n · μ. Thus the basic equilibrium equation

$$p(i) \cdot \lambda_i = p(i+1) \cdot \mu_{i+1}$$

applied to the two regions $0 \le i \le n$ and $n \le i \le \infty$ will have slightly different parameters:

$$\lambda_i = \lambda,\ \mu_i = i \cdot \mu \text{ for } 0 \le i \le n, \text{ and} \qquad (75)$$
$$\lambda_i = \lambda,\ \mu_i = n \cdot \mu \text{ for } n \le i \le \infty$$

The analysis based on these assumptions is given in Frame 6.

Characteristics of the M/M/n queuing system as presented in Frame 6 is given in concise form in Table 4.

The term $(n - A)$

As may be observed in Table 4, all formulas except the single state probabilities contain the term $(n - A)$. This is a remarkable and general characteristic of a queuing system. It indicates an asymptotic behaviour, as the denominator in queue length and waiting time expressions tends to zero, and hence the expression itself tends to infinity whenever $A \to n$.

Since the stability condition is $A < n$, and A is the mean value of a stochastic variable, the difference $n - A$ expresses a surplus capacity, also called the *traffic reserve* of the system. In theory this reserve can be fully utilised, but only at the cost of an infinite queue and unlimited waiting time.

15.2 A general relation for a no-loss queuing system

For any queuing system G/G/n we know that Little's formula applies. That means that when we have calculated waiting times and queue lengths averaged over all calls, we can directly find total time *(T)* and total number *(Λ)* in system by adding service time *(s)* to waiting time *(W)* and mean load *(A)* to queue length *(L):*

$$T = W + s \tag{87}$$

$$\begin{aligned}\Lambda &= \lambda \cdot T = \lambda \cdot (W + s)\\ &= \lambda \cdot W + \lambda \cdot s = L + A\end{aligned} \tag{88}$$

16 Single server queues

16.1 The M/M/1 queue

By introducing $n = 1$ for the M/M/n system above, we have the single server Markov queue M/M/1:

Waiting time distribution:
$F_w(t) = 1 - A \cdot e^{-\mu(1-A)\cdot t}$

Sojourn time distribution:
$F_s(t) = 1 - e^{-\mu(1-A)\cdot t}$

Waiting probability:
$E_2 = A < 1$

Mean number in queue:
$L = A^2/(1 - A)$

Mean number in system:
$L + A = A/(1 - A)$

Mean waiting time:
$W = s \cdot A/(1 - A)$

Mean sojourn(response) time:
$T = W + s = s/(1 - A)$

For comparison the M/M/1 loss system has a loss probability of $E_1 = A/(1 + A)$. If e.g. $A = 0.9$, then $E_1 = 0.47$ and $E_2 = 0.9$. This means that nearly half of the calls are lost in the loss system, while in the waiting system 90 % will have to wait. The average queue length will be 8.1 and mean response time $10 \cdot s$. Thus the important parameter for a waiting person is the length of the service time s. The response time in this case is ten times the service time. However, if the service time is, say, 10 milliseconds, the response time is hardly noticeable. A service time of 1 minute, on the other hand, would leave the person waiting for 10 minutes! This illustrates how the service time is a very crucial parameter under otherwise equal load conditions, and it also indicates important reasons for the choice between loss and waiting.

16.2 The M/G/1 queue

The M/G/1 queue is the one-server case with Poisson arrivals with parameter λ and a mean service time s with any distribution. There is no simple explicit expression of the queue and waiting time distributions, but there are fairly simple Laplace transform expressions. For the waiting time distribution we have:

$$\begin{aligned}F_w*(s) &= L\{F_w(t); s\}\\ &= \frac{1-A}{s - \lambda[1 - f_s*(s)]}\end{aligned} \tag{89}$$

where $f_s*(s)$ is the Laplace transform of the service time distribution density function. The mean value of the waiting time in the general case can be determined by differentiation of $f_w*(s) = s \cdot F_w*(s)$. This differentiation leads to a 0/0-expression. By L'Hopital's rule applied twice (differentiating numerator and denominator separately), the result is obtained

$$W = A \cdot \frac{\varepsilon}{2\mu(1-A)}$$

$$\varepsilon = \frac{f_s*''(0)}{f_s*'(0)^2} = M_2/M_1^2$$

In the M/M/1 case we have

$$f_s*(s) = \frac{\mu}{s+\mu}$$

leading to

$$F_w*(s) = (1-A)\cdot\frac{1+\frac{\mu}{s}}{s+\mu(1-A)}$$

with the inverse transform

$$F_w(t) = 1 - A \cdot e^{-\mu(1-A)\cdot t} \tag{90}$$

in accordance with the previous result for M/M/1.

A direct basic approach (without the use of transforms) can be used to find mean values. Little's formula is again a useful tool. The mean waiting time for an arbitrary arrival can be determined as the sum of remaining service for the customer being served and the full service times of those ahead in the queue. If the system is empty, there will be zero waiting time, while in a non-empty system the remaining time for the customer in service (for any distribution) is given by $\varepsilon/2\mu = (c^2 + 1)/2\mu$. (See section 10.1). Weighted by the probability of a non-empty system, A, this gives an average contribution $A \cdot \varepsilon/2\mu$. The number of customers in the queue is, according to Little's formula, $L = \lambda \cdot W$. Each of those in queue will need on average a service time $s = 1/\mu$, leading to $L \cdot s = \lambda \cdot W \cdot s = A \cdot W$. Thus, the overall mean waiting time can be expressed by the implicit equation

$$W = \frac{A \cdot s \cdot \varepsilon}{2 + A \cdot W} \Rightarrow W = \frac{A \cdot s \cdot \varepsilon}{2 \cdot (1 - A)} \tag{91}$$

The formula is known as the Pollaczek-Khintchine formula. Since $\varepsilon = 2$ for an exponential distribution, we obtain our previous result $W = A \cdot s / (1 - A)$. The minimum value $\varepsilon = 1$ applies to the deterministic distribution, with the result $W = A \cdot s/[2 \cdot (1 - A)]$. For a hyperexponential distribution with parameters $p = 0.9$, $\lambda_1 = 1$ and $\lambda_2 = 1/11$ (to give $\lambda = 0.5$) we have $\varepsilon = 6.5$, which implies a waiting time 3.25 times that of an exponential distribution and 6.5 times that of a deterministic distribution.

The mean queue length of the M/G/1 system is given straightforward by Little's formula

$$L = \lambda \cdot W = \frac{A^2 \cdot \varepsilon}{2 \cdot (1 - A)}$$

16.3 The GI/M/1 queue

In this case the queue behaviour is identical to that of the M/M/1 case, while the arrivals behave differently. The technique of embedded Markov chains is a useful tool, as one can focus on the system state immediately before an arrival instant. If the probability of the server being busy at that instant is σ, then the corresponding state probability is given by the geometric distribution $p(i) = (1 - \sigma) \cdot \sigma^i$, (with the mean value $\sigma/(1 - \sigma)$). Since the service is Markovian, we can use the same type of calculation for the waiting time distribution as that of the M/M/n system for a product sum of Erlang and geometric distributions, to give

$$\begin{aligned}&F_w(t) = 1 - e^{-\mu(1-\sigma)\cdot t};\\ &G_w(t) = 1 - F_w(t) = e^{-\mu(1-\sigma)\cdot t}\end{aligned} \tag{92}$$

Since $G_w(t)$ is the survivor function, it means the probability of the last previous arrival still being in the system. If the arrival interval distribution is given by density $f(t)$, then $P\{t < \text{arrival interval} \leq t + dt\} = f(t) \cdot dt$. The product of the two independent terms, $G_w(t) \cdot f(t)dt$, implies that the next arrival happens in the interval $(t, t + dt)$, and that there is still one or more customers in the system. By integration over all t we obtain the implicit equation for determination of σ:

$$\sigma = \int_0^\infty e^{-\mu(1-\sigma)\cdot t} \cdot f(t)dt \qquad (93)$$

It is immediately seen that if $f(t) = \lambda e^{-\lambda t}$, then $\sigma = \lambda/\mu$, and we get the M/M/1 solution as we should.

By analogy it is clear that the mean values of waiting and sojourn times, which are related to calls arriving, and queue lengths as observed by arriving calls, are given by formulas identical to those of M/M/1, only that the mean load A is replaced by the arrival related load σ. Taken over time the mean values of number in queue (L_q) and number in system (L_s) are found by Little's formula, based on mean waiting time:

$$L_q = \lambda \cdot W = \frac{\lambda \cdot s \cdot \sigma}{1-\sigma} = \frac{A \cdot \sigma}{1-\sigma}$$

$$L_s = \lambda \cdot (W + s) = \frac{\lambda \cdot s}{1-\sigma} = \frac{A}{1-\sigma} \qquad (94)$$

Example

In Table 5 is given an example set with arrival interval distributions:

- Deterministic
- Erlang-2
- Exponential
- Hyperexponential (H_2).

For the H_2-distribution is chosen $p = 0.9$, $\lambda_1 = 1.0$ and $\lambda_2 = 1/11$, like in the M/G/1 example previously, and two more cases: $p = 0.75/0.95$, $\lambda_1 = 1.0/1.0$ and $\lambda_2 = 0.2/0.0476$. Arrival rate is chosen at $\lambda = 0.5$ (consistent with the H_2-examples) and service time $s = 1/\mu = 1.0$. Mean load is $A = \lambda/\mu = 0.5$, and service is exponential.

The examples in the table indicate that waiting time and queue length increase with the form factor, but in a non-linear way. This is different from the M/G/1 case, where there is strict proportionality.

16.4 The GI/GI/1 queue

When no specification is given for the distributions of arrival and service processes, other than independence within each process and mutually, then there are few general results available concerning queue lengths and waiting time. A general method is based on Lindley's integral equation method. The mathematics are complicated, and it will not be discussed any further. There are a lot of approximations, most of them covering a limited area with acceptable accuracy. According to the Pollaczek/Khintchine formula for M/GI/1 two moments of the service distribution gives an exact solution. The GI/M1 case is shown to be less simple, and even though most approximations for the GI/GI/1 queue apply only two moments for both distributions, it is clear that even the third moment of the arrival distribution may be of importance.

According to [15] a reasonable approximation for high loads ($A \rightarrow 1$) is the one by Kingman [16], which is also an upper limit given by

$$W \approx \frac{A \cdot s}{2 \cdot (1-A)} \cdot \left\{ (c_a/A)^2 + c_s^2 \right\} \qquad (95)$$

where c_a and c_s are coefficient of variation for arrivals and service respectively.

We immediately recognise the basic terms giving proportionality with load (A) and service time (s) and the denominator term $2(1 - A)$. Next is the strong dependence on second moments. The formula approaches the correct value for M/M/1 when $A \rightarrow 1$. Otherwise it is not a good approximation. An adjustment factor has been added by Marchal [17]:

$$(1 + c_s^2)/(1/A^2 + c_s^2) \qquad (96)$$

This gives the correct result for M/G/1 irrespective of load.

A better adjustment to Kingman's formula has been introduced by Kraemer and Langenbach-Belz [18], using an exponential factor containing the load as well as the coefficients of variation. Distinct factors are used for the two regions $c_a \leq 1$ and $c_a \geq 1$. The best general approximation over a wide range seems to be that by Kimura [19]:

$$W \approx \frac{\sigma \cdot s \cdot \left(c_a^2 + c_s^2\right)}{(1-\sigma) \cdot (c_a^2 + 1)} \qquad (97)$$

Here, σ is the mean load immediately before an arrival, like the one in the GI/M/1 case, given by an integral equation. Thus the real distribution (not only first and second moments) is taken into account. Calculation is not straightforward because of the implicit solution.

Another proposal [20], [21] is based on a three moment approach for arrivals, applying a three-parameter H_2 distribution. The three-moment match is much more flexible than the two-moment match, and it can be used to approximate a general distribution of the type where $c_a \geq 1$. The approximation utilises the property of the H_2 distribution that it can model two extremes by proper choice of parameters:

1 Batch Poisson arrivals with geometric batch size

2 Pure Poisson arrivals.

Exact solutions are available for the whole range of H_2/M/1 between the extremes M^b/M/1 and M/M/1. In the general service case the extreme points M^b/GI/1 and M/GI/1 have known exact solutions. A heuristic interpolation between those two points can then be attempted for the GI case in analogy with the known M case. A plot for the waiting time as a function of the third moment is shown in Figure 37.

Table 5 Example with mean load A = 0.5 of GI/M/1 system

Distribution	Load at arrival *(σ)*	Waiting time *(W)* = Queue length *(L)/λ*	Form factor *(ε)*
Deterministic	0.2032	0.255	1.0
Erlang-2	0.382	0.618	1.5
Exponential	0.5	1.0	2.0
Hyperexponential-2	0.642	1.79	3.5
"	0.74	2.85	6.5
"	0.804	4.11	11.5

The relative moments $r = M_2/2M_1^2 = \varepsilon/2$ and $q = M_3/6M_1^3$ are chosen so that $r = q = 1$ in the exponential case. The two extremes occur when $q = q_o = r^2$ and $q = q_\infty = \infty$. $q_o = r^2$ is the smallest possible value of q.

The M^b/GI/1 and M/GI/1 waiting times are given, respectively, by

$$W = \frac{A}{2\mu \cdot (1 - A)} \cdot \{\varepsilon_s + (\varepsilon_a - 2)/A\}$$

and

$$W = \frac{A}{2\mu \cdot (1 - A)} \cdot \varepsilon_s$$

where ε_a and ε_s are arrival and service form factors. (It may be noted that the asymptotic form of the upper formula when $A \to 1$ is identical to that of Kingman's.) The obvious approach is to find a multiplier function varying between 1 and 0 for the second term in the M^b form. This function would naturally contain at least the relative third moment q and possibly r and A, $f(q, r, A)$. The waiting time approximation based on this approach will have the form

$$W \approx \frac{A}{2\mu \cdot (a - A)} \cdot \{\varepsilon_s + f(q, r, A) \cdot (\varepsilon_a - 2)/A\} \qquad (98)$$

Suggested forms of $f(q, r, A)$ are given in [20] and [21], along with simulation results and comparisons with other approximations. The effect of the third moment is not included in any of the other approximations, except for Kimura's, where it is implicitly accounted for by the determination of σ. As an example of the third moment effect one can simply compare the two limiting formulas above with identical values of A, ε_a and ε_s. The extreme q-values are $q_o = (\varepsilon_a/2)^2$ and $q_\infty = \infty$. As an illustration we see that if the load is $A = 0.5$, the service is deterministic, $\varepsilon_s = 1$, and the arrival distribution has a coefficient of variation $c_a = 2(\varepsilon_a = 5)$, then we obtain the striking result for the third moment influence that

$W(q_o) = 7 \cdot W(q_\infty)$ (!)

16.5 Queue disciplines

Throughout the previous considerations we have assumed the queuing principle of first in – first out, or first come – first served (FIFO = FCFS). That seems a fair principle at the outset. There are, however, good arguments for the application of other principles. Last in – first out (LIFO), also called the stack principle, may have advantages in technical solutions. One example is in an overload situation to apply LIFO to push out those who have waited longest, and thus bring down the mean waiting for those who succeed. Service in random order (SIRO) may help circumventing faulty equipment and equalise mechanical wear. Mean waiting time is independent of the mentioned disciplines, since the sequence is not in any way dependent on the service time, whereas the distributions are different.

From everyday life we may have observed, and even experienced, that a customer with a huge basket of goods arrives at the counter just ahead of the next one carrying only a loaf of bread. If the first lets the second bypass him, he loses only a moment, whereas the other saves a lot of time. This is an example of the discipline of shortest job first (SJF). Clearly, the job of the clerk at the counter is the same, still the mean waiting time is reduced. The explanation of this little paradox is that each customer represents different load on the server (clerk). Say that the first customer has $n_1 = 20$ pieces against just $n_2 = 1$ piece for the second customer. The swapping between them reduces the total waiting time from 0 + 20 = 20 to 0 + 1 = 1. However, weighting the waiting time by the number of pieces leads to a change from $20 \cdot 0 + 1 \cdot 20 = 20$ to $1 \cdot 0 + 20 \cdot 1 = 20$, i.e. the weighted result is unchanged by the swapping.

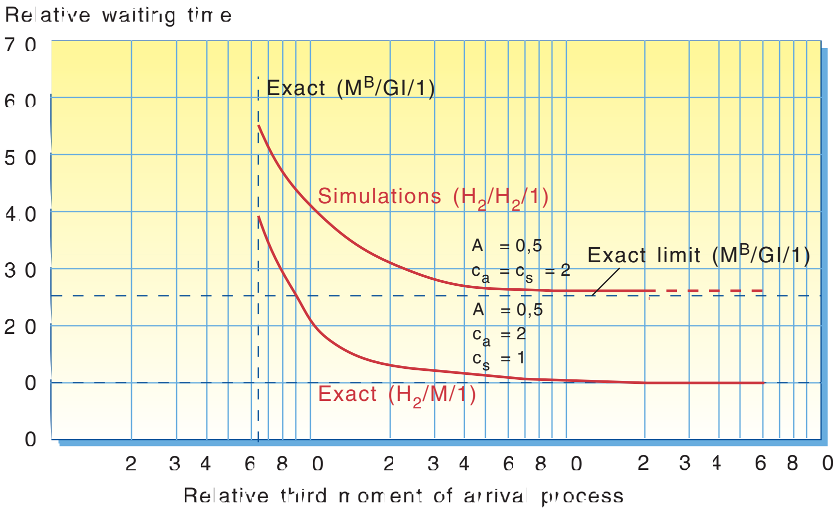


Figure 37 The influence of the third moment of the arrival interval distribution

If three messages of lengths 1, 3 and 10 are to be transmitted, we can compare the mean waiting times and find a strong sequence dependence, whereas the weighted means are identical:

$m3$=10, $m2$=3, $m1$=1

Mean wait: $w = (0 + 1 + 4)/3 = \underline{5/3}$
Weighted mean: $1 \cdot 0 + 3 \cdot 1 + 10 \cdot 4 = \underline{43}$

$m1$=1, $m2$=3, $m3$=10

Mean wait: $w = (0 + 10 + 13)/3 = \underline{23/3}$
Weighted mean: $10 \cdot 0 + 3 \cdot 10 + 1 \cdot 13 = \underline{43}$

The SJF discipline reduces the total waiting time to a minimum, while the system load is unchanged. Queue disciplines and priorities are studied to a great extent by Kleinrock [14], and further studies are referred to that source.

A particularly interesting case is that of Round Robin (RR), with the limiting case Processor Sharing (PS). Customers arrive in a FIFO queue, but instead of finishing each job when it is being served, only a fixed element of service time is allotted, whereafter the job is returned to the queue input. Thus the number of times a job is given a fixed element of service is proportional to the job size. The result is identical to that of a parallel service, where each of k customers has at his disposal $1/k$ of the processor capacity. PS has some nice properties:

- Response time is proportional to the service time of each job (fairness). Hence also waiting time is proportional to service time.

- Mean response time depends only on mean value (not higher order moments) of service time
- For exponential service times a job with mean service time will have response time equal to that of a batch system. Shorter jobs come out better and longer jobs worse in PS than in batch. In this sense PS comes between batch and SJF.

17 Queuing network analysis

Up till now we have considered only single nodes, whether loss systems or queues. In this article we have no ambition of making any thorough investigation into networks, as it is a rather complicated matter that would far exceed the present aim.

A network model consists of nodes and links. In traditional networks signal propagation time is of no particular concern. Radio propagation has a speed of 300,000 km/s, and for cable transmission the assumption is roughly 200,000 km/s. Thus the one-way delay via a geostationary satellite is 0.25 seconds, and that of a cable transmission halfway around the globe 0.1 seconds. Compared to telephone set-up and conversation times those delays are short, whereas in a real time dialogue the 0.25 second delay is critical. The delay on a 500 m local area network is 2.5 µs, corresponding to transmission of 25 bits with 10 Mb/s transmission speed. With a CSMA/CD access system even that short delay is of significance. The interesting parameter is the Propagation time – to – Transmission time ratio: $T_p/T_t = a$. Propagation time is a fixed value irrespective of signal type, whereas transmission time is proportional to frame length in bits and inversely proportional to the bit rate.

For simple one-way transmission messages may be chained continuously at the sending end, and a utilisation of $U = 1$ can be obtained. If, on the other hand, after one transmission a far end user wants to start sending, the utilisation will be maximum

$$U = 1/(1 + a) \qquad (99)$$

In data communication flow control is often performed by acknowledgement, adding one propagation time (plus a negligible transmission time) to one transmission time and one propagation time in the forward direction. Thus the gross time used per transmitted information frame is $T_t + 2T_p$, and the utilisation (transmission efficiency) is

$$U = T_t/(T_t + 2T_p) = 1/(1 + 2a) \qquad (100)$$

Transmission of a 1 kb frame with acknowledgement at 64 kb/s over a 10 km cable gives a utilisation $U = 0.994$, whereas the same communication over a satellite leads to $U = 0.06$.

In these estimates we have assumed that all transmitted data is useful information. There will always be a need for some overhead, and the real utilisation will be reduced by a factor $I/(I + O)$, where I = information and O = overhead.

17.1 Basic parameters and state analysis

The discussion above is included to give a realistic sense of the propagation delay importance. In the following approach we assume that the signal delay may be neglected, so that a utilisation $U = 1$ can be obtained. In data communication networks information messages are usually sent as serial bitstreams. Messages may be sent as complete entities, or they may be subdivided in frames as sub-entities sent separately. The time that a link is occupied, which is the holding time in a traffic context, depends on the transmission capacity C. In a similar way a computer may be considered a server with a certain capacity C. C may be measured arbitrarily as operations per second or bits per second. We can define a service time by $1/\mu C$, being the time to complete a job, whether it is a computer job or a message to be transmitted. Examples are

1 C: operations/second
$1/\mu$: operations/job
μC: jobs/second

2 C: bits/second
$1/\mu$: bits/message
μC: messages/second

If we stick to the second example, μC means the link capacity that is obtained by chaining messages continuously on the link. Assuming that the messages arrive at a rate $\lambda < \mu C$, we obtain an average load on the link $A = \lambda/\mu C$. Note that the product μC here replaces μ in previous contexts. In a data network the dynamic aspect of a flow of messages along some path is often emphasised. The number of messages per second, λ, is often termed the *throughput.* The link load A expresses the number of messages arriving per message time, and is thus a *normalised throughput*.

Assume a network of N nodes and M links. The total number of message arrivals to (= number of departures from) the network per second is

$$\gamma = \sum_{j=1}^{N} \sum_{k=1}^{N} \gamma_{jk} \qquad (101)$$

where γ_{jk} is the number of messages per second originated at node j and destined for node k. For an arbitrary node i the total number of arrivals per second (external and internal) is

$$\lambda_i = \gamma_i + \sum_{j=1}^{N} \lambda_j \cdot p_{ji} \qquad (102)$$

where $p_{ji} = P\{$a customer leaving j proceeds to $i\}$. Since a customer can leave the network from any node,

$$P\{\text{departure from network at node } i\} = 1 - \sum_{j=1}^{N} p_{ij}$$

If each node i is an M/M/n_i system, departures from each node is Poisson, and the nodes may be analysed independently. This leads to the important property of *product form* solution for the network, which means that the joint probability of states k_i can be expressed as the product of the individual probabilities:

$$p(k_1, k_2, \ldots, k_N) = p_1(k_1) \cdot p_2(k_2) \cdot \ldots p_N(k_N) \qquad (103)$$

The solution for $p_i(k_i)$ is the one developed for the M/M/n waiting system, equations (76):

$$p(i) = (A/i) \cdot p(i-1) = (A^i/i!) \cdot p(0), \quad 0 \le i \le n$$

and

$$p(i) = (A/n) \cdot p(i-1) = (A^n/n!) \cdot (A/n)^{i-n} \cdot p(0), \quad n \le i \le \infty$$

17.2 Delay calculations

Delay calculations are fairly simple for the network of M/M/1 nodes. The sojourn time at a node i is expressed by

$$T_i = 1/(\mu C_i - \lambda_i) \qquad (104)$$

The average message delay in the network can be expressed by [14]:

$$T = \frac{\sum_{i=1}^{M} \lambda_i T_i}{\gamma} = \frac{\sum_{i=1}^{M} \lambda_i}{\gamma(\mu C_i - \lambda_i)} \qquad (105)$$

Since the calculated delays are those of the M/M/1 sojourn times, node processing times and link propagation delays should be added to give complete results.

18 Local area networks

Up till now the traffic system description and analysis has assumed very simple methods of access to the available servers. That is assured by having only one access point at each end of a link. Simultaneous occupation from both ends is resolved before any user information is transmitted. Competition for resources is resolved by rejection of attempts or by queuing. In multi-access systems competition can be handled by means of polling by a control node, by handing over sending permission (token) from node to node, or simply by trying and (if necessary) repeating.

18.1 The ALOHA principle

The simplest assumption for transmission over a common channel, whether it is a frequency band for radio transmission or a local bus, is the one called ALOHA. The normal Poisson case of an unlimited number of independent sources each with an infinitesimal call rate to give a resulting finite total call rate is assumed. Each transmission is assumed to last one time unit. There is a common control node to detect collisions between transmissions, and all sources have the same delay to the control node. Collisions result in retransmissions. At very low offered rates, collisions are very unlikely, and throughput will be equal to offered traffic. With growing call rates the number of collisions will increase, thus increasing the number of retransmissions, and after passing a maximum the throughput will decrease with increasing call rate, until virtually nothing is let through. When one packet is being transmitted, a packet from another station will partially overlap if transmission starts within a time window twice the packet length.

A simple deduction leads to the formula for throughput

$$S = Ge^{-2G} \tag{106}$$

where G = number of transmitted packets per time unit. That includes primary transmissions and retransmissions as well. We see that $S' = dS/dG = e^{-2G}(1 - 2G) = 1$ for $G = 0$, and $S' = 0$ for $G = 0.5$ and $G = \infty$. Thus, $S_{max} = 0.5/e = 0.18$.

An improvement is obtained by synchronising transmission in time slots to give slotted ALOHA. Thus collisions only happen in complete overlaps, and the critical period is one time unit instead of two, to give

$$S = Ge^{-G} \tag{107}$$

with $S_{max} = 1/e = 0.37$ for $G = 1$.

18.2 The CSMA/CD principle

The ALOHA analysis can be modified by taking into account delays. When the simple form is used, it can be supported by the fact that it has been used mostly with low rate radio transmission, where the propagation delay is short compared with the packet transmission time ($a = T_p/T_t << 1$).

In high capacity networks the propagation time is of great importance, even at short distances. The CSMA/CD (Carrier sense multiple access/Collision detection) principle is based on a station listening to the common bus when it wants to send, and to back off when busy. Collision detection is used to stop sending immediately. A station A at one end of the bus starts sending a packet that reaches a station B at the other end after a time T_p. Just before this instant station B still finds the bus free and starts sending. Station A detects the collision at $2T_p$ after it started sending. We assume the time divided in slots of length $2T_p$ and the probability of exactly one station attempting a transmission in a slot is Q. $(1 - Q)$ then expresses the probability of either no attempt or a collision. The average number of time slots before a successful transmission is given by the geometric distribution $(1 - Q)^i \cdot Q$ with mean $(1 - Q)/Q$. The normalised throughput now can be expressed by

$$S = \frac{T_t}{T_t + 2T_p(1-Q)/Q} = \frac{1}{1 + 2a(1-Q)/Q} \tag{108}$$

The maximum of Q is obtained from the binomial distribution for equal load by all N stations

$$Q_{max} = (1 - 1/N)^{N-1} \rightarrow 1/e$$

to give

$$\lim_{N \to \infty} S = \frac{1}{1 + 2a(e-1)} \tag{109}$$

Examples:
$a = 1.0$, $\lim S = 0.22$; $a = 0.1$, $\lim S = 0.74$; $a = 0.01$, $\lim S = 0.97$

A lower number of stations gives a higher throughput with the same offered load. Q, and thus the throughput, will be reduced with a lower offered load because of more unused slots or with a higher offered load because of more collisions.

18.3 The Token Ring principle

Sending permission is distributed in a fair way by passing a token sequentially from station to station along a ring. We do not discuss the many variants of token ring (priorities, multiple tokens etc.). In the simple version, a station A receives a free token, sets the token busy and puts it at the head of a message to some station X. After reception, X returns the message with busy token along the ring back to A as an acknowledgement. As soon as A has finished sending *and* has received the busy token, A passes it on as a free token, which is picked up by the first station, say B, that has something to send.

As before we assume a message transmission time T_t and a full ring propagation time T_p, with $a = T_p/T_t$. Maximum throughput is obtained when all stations have messages ready to send. Equal time distance along the ring is assumed: T_p/N. Two cases occur:

$$a \le 1 : S = \frac{T_t}{T_t + \frac{T_p}{N}} = \frac{1}{1 + \frac{a}{N}}$$

$$a \le 1 : S = \frac{T_t}{T_p + \frac{T_p}{N}} = \frac{1}{a(a + \frac{1}{N})} \tag{110}$$

We see that the throughput for Token Ring as well as for CSMA/CD gains by a low $a = T_p/T_t$-ratio. A large number of stations has opposite effects, with an advantage for Token Ring, but the main distinction is that of collision or no collision. Thus Token Ring may well obtain a throughput (utilisation) of > 0.9, whereas CSMA/CD would experience congestion at a much lower point.

19 Mixed traffic and high capacity systems

The main focus of the article has been basic traffic relations and models based initially on telephone traffic relations. However, also data traffic has been discussed, and the distinct burstiness of such traffic has been pointed out. (See for instance Figure 10.) There is far less knowledge available on real data traffic than that of telephone traffic. The two main reasons for this are 1) the long tradition (~ 90 years) of studies, and 2) the

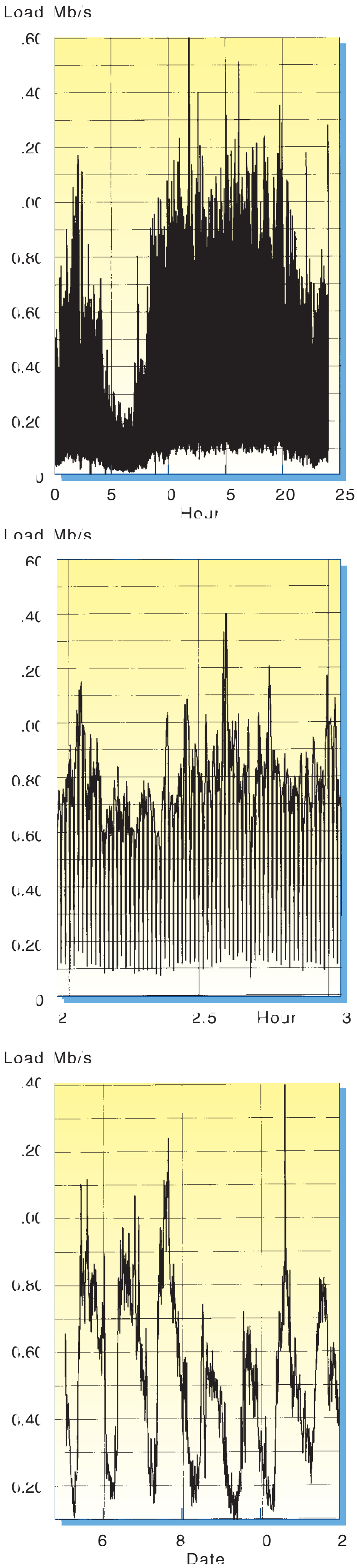


Figure 38 Mixed international data traffic on an inter-university network

much greater regularity of the latter. One should also point out the much greater variety in transmission speeds of data traffic. Telephone traffic uses a normalised bit rate of 64 kb/s, and it is of real time dialogue type. Data traffic occurs over a scale of some 7 decades (50 bits/s – 500 Mb/s) and increasing; usually the real time requirements are much more relaxed. Telephone conversation is sensitive to overall delays above 250 milliseconds. Geostationary satellites alone introduce 250 ms propagation delay, whereas terrestrial global communication stands for up to 100 ms. The 48 byte cell structure of ATM introduces 125 μs x 48 = 6 ms. Delay per node should be in the order of magnitude of no more than 1 ms. The corresponding sensitivity for most data types may vary from a few hundred milliseconds to hours! A particular consideration is that of synchronising the services in a multi-service communication, leading to stricter delay limits. With strict synchronisation as a requirement the slowest service (e.g. video pictures) is determinant, and may lead to inconvenient voice delay.

Most networks up till now have been *service dedicated* networks. Notorious are the fixed rate switched telephone network on one side and the non-switched distribution networks for sound (radio) and video (television) on the other. (The pre-scheduled, pure distribution networks of radio and TV are less interesting from a traffic point of view.) The partly fixed and partly variable bit rate dedicated data networks still cater for much lower traffic volumes than the former two, as measured in bits/s.

Some studies [22] seem to indicate that with mainly the present types of services it will take many years until data traffic approach telephone traffic in aggregated volume as measured by bits/s. The types of services that will have to take off in order to change this picture are widespread video telephones and video-on-demand in the residential sector and large file transfers for high resolution graphics and video in the business sector. A distinct development is that of high rate bursty communication to the desktop, requiring big bandwidth to the user.

There are great potential gains in integration of different service types for the following reasons:

- common infrastructure
- better utilisation caused by volume (Erlang's formulae)
- smoothing by mixing of traffics
- shaping, control and management means in new high capacity systems.

Against these advantages count scale disparities that tend to offset the single service optimums.

19.1 Classification of high rate data types

Studies within RACE projects [23] have identified five application types in integrated high capacity systems (B-ISDN):

- file transfer without real time requirements (text, image, video, audio)
- remote access interactive retrieval functions with high demands on delay and synchronisation (incl. video)
- joint editing/viewing with annotation, transmission of changes, synchronised video/audio
- bursty communication with high demand for flexible access and bandwidth
- widespread access, typically for video-on-demand and videotelephony.

An example of data traffic is previously given in Figure 10. Other examples are shown in [24]. Some data for traffic on a 2 Mb/s link of the academic network UNINETT are shown in Figure 38. The traffic is typically a mixture of electronic mail, remote access, file transfer, info search and broadcast. The last two services are the more recently developed services on WWW (World Wide Web).

Another example in Figure 39 shows observation results from SUPERNETT, an experimental inter-university/research institute national network operating at 34 Mb/s. Also here various traffic types are mixed. However, there is no international traffic, and super-computer remote access is included.

For simplicity only one-way traffic (outgoing Trondheim – Oslo) is plotted. It should be noted that the plots are random samples, though they seem to give a typical picture. The much greater bitrate variation of the high-speed link is notable. The 24-hour and 1-hour profiles are based on a resolution of 10 seconds, whereas the week profiles use 15 minute resolution. Both examples are supplied by the UNINETT A/S company.

19.2 Traffic management in high capacity systems

The agreed protocol of high capacity transmission systems for B-ISDN is ATM (Asynchronous Transfer Mode). The intention has been to obtain a ubiquitous format to cover the needs of all types of services, whether high or low speed, synchronous or non-synchronous, one-way or dialogue, real time, interactive or file transfer.

The performance measures of the cell-based transmission system are related to the *call connection phase* or the *information transfer phase*. In the call connection phase the main criteria are rejection (blocking) probability and delays of set-up, release and in-call changes. In the information transfer phase the main criteria are error, loss or misinsertion of cells and cell delay and delay variations.

In an environment with many high-speed devices having access to the network there is the risk of some users monopolising the available capacity. It is, therefore, necessary to use some admission control according to a *contract,* and after the admission carry out a control that the contract is adhered to, employing some kind of *police* function.

The distinction between VBR (variable bit rate) and CBR (constant bit rate) is important. For CBR the *peak cell rate* is the most important parameter, while for VBR the *sustainable cell rate* is of prime interest.

A particular problem in ATM and similar multi-rate systems is the inequitable blocking of high-rate and low-rate transmission. The latter will find enough capacity very often while the former is blocked. Since in practice the bandwidth needs tend to be extremely different (several decades), one solution is to dedicate separate resources to larger rates and common resources to be shared by all kinds of demands. The main objective may not be full fairness among demands, but to keep blocking below specified limits and maintain a high utilisation.

To improve the performance, *traffic shaping* is applied to reduce peaks where the type of traffic (e.g. VBR) permits such influence.

Traffic analysis of a multi-service system like ATM is quite complicated. The analysis may focus on different levels: *cell* level, *burst* level or *call* level. There is a great number of publications on the subject. Methods of analysis are discussed in [25] in this issue. The limitation of analytical methods is pointed out in [26], claiming that measurements are indispensable, and artificial traffic generation for that purpose is described. Also simulations encounter particular problems. The strict cell loss requirements (10^{-9} – 10^{-11} cell loss ratio) lead to very long simulation runs in order to obtain acceptable confidence. Speed up methods are discussed in [27].

20 Concluding remarks

The main objective of the present article is to offer an introduction to the vast area of teletraffic problems in a way that makes it readable without requiring a deep prior knowledge of the subject. Even the indispensable instrument of mathematics is kept at what is deemed a minimum in extension as well as level of difficulty. Some extension for those interested is provided within separate frames.

It is sensible to keep in mind that on one side this basic presentation lacks some mathematical strictness, when seen from a mathematician's fundamental point of view. On the opposite side there is the area of applications, where even much more is lacking. It may suffice to mention modelling of various systems and the corresponding statistical distributions, methods of analysis, simulations, matching methods, dimensioning principles, measurement methods and corresponding theory, statistical confidence, prognosis, as well as all the new aspects connected with the rapid development of technology. A serious treatment of all those aspects would require many volumes, which is in fact the case, considering the extensive specialised literature of the field.

In spite of these reservations it is the belief of this author that the article may be applied as a stand-alone presentation of the subject of teletraffic, as well as a useful introduction that can support further studies.

21 References

1 Iversen, V B. *Data- og teletrafikteori.* Den Private Ingeniørfond 1985.

2 Parviala, A. Observed traffic variations and their influence in choice of intensity measurement routines. *Telektronikk,* 91(2), 69–78, 1995. (This issue.)

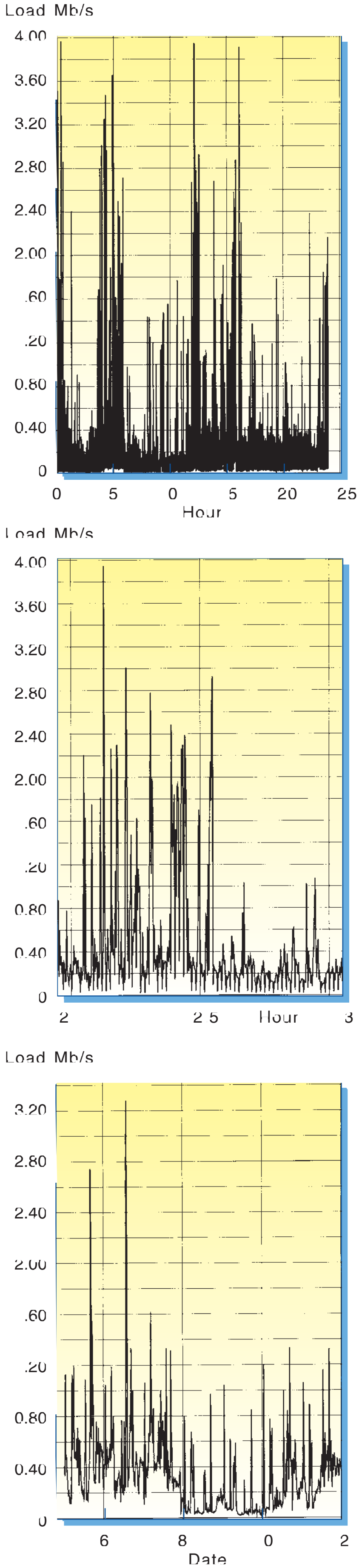


Figure 39 Mixed national data traffic on an experimental high-speed network

3 Gaaren, S. *LAN interconnection traffic measurements : the methodology.* Kjeller, Televerkets Forskningsinstitutt, 1994. (TF report R 51/94.)

4 Khintchine, A Y. *Mathematical methods in the theory of queuing.* London, Griffin, 1969.

5 Myskja, A, Walmann, O O. A statistical study of telephone traffic data with emphasis on subscriber behavior. I: *7th international teletraffic congress, ITC 7.* Stockholm 1973.

6 Myskja, A. A recording and processing system for accounting and traffic analysis on a large PABX. *IEEE transactions on communication technology,* COM-19(5), 1971.

7 Gaustad, O. *Registrering, analyse og anvendelse av trafikkdata fra telefonnettet.* Trondheim, Institutt for teletrafikksystemer, Universitetet i Trondheim, Norges tekniske høgskole, 1974.

8 Feng, B, Myskja, A, Heegaard, P E. *Analyse av ekstern trafikk ved UNIT/SINTEFs telesentral, MD110.* Trondheim, SINTEF DELAB, 1992. (STF40 A92047.)

9 Bolotin, V A. Telephone circuit holding time distributions. I: *14th international teletraffic congress, ITC 14,* Elsevier, Amsterdam, 1994.

10 Wilkinson, R I. Theories for toll traffic engineering in the USA. *Bell system technical journal,* 35, 1956.

11 Kuczura, A. The interrupted poisson process as an overflow process. *Bell system technical journal,* 52, 1973.

12 Wallstrøm, B, Reneby, L. On individual losses in overflow systems. I: *9th international teletraffic congress, ITC 9,* Torremolinos, 1979.

13 Wallstrøm, B. Point-to-point losses in hierarchical alternative routing. *Telektronikk,* 91(2), 79–81, 1995. (This issue.)

14 Myskja, A, Aagesen, F A. On the interaction between subscribers and a telephone system. I: *8th international teletraffic congress, ITC 8,* Melbourne, 1976.

15 Kleinrock, L. *Queuing systems, vol II : computer applications.* New York, Wiley, 1976.

16 Kingman, J F C. On queues in heavy traffic. *Journal of the royal statistical society,* Series B, 24, 1962.

17 Marchal, W G. An approximate formula for waiting time in single server queues. *AIIE transactions,* 8, 1976

18 Kraemer, W, Langenbach-Belz, M. Approximate formulae for the delay in the queuing system GI/G/1. I: *8th International teletraffic congress, ITC 8,* Melbourne, 1976.

19 Kimura, T. Refining diffusion approximations for GI/G/1 queues : a tight discretization method. I: *11th international teletraffic congress, ITC 11,* Kyoto, 1985.

20 Myskja, A. On approximations for the GI/GI/1 queue. *Computer networks and ISDN systems,* 20, Elsevier, 1990.

21 Myskja, A. An improved heuristic approximation for the GI/GI/1 queue with bursty arrivals. I: *13th international teletraffic congress, ITC 13,* Copenhagen, 1991.

22 Noll, A M. Voice vs data : an estimate of future broadband traffic. *IEEE communications magazine,* June 1991.

23 Roy, M, de Sousa, P. Integrated broadband communications in Europe. *Telecommunications,* Nov. 1994.

24 Gaaren, S. LAN interconnection traffic measurements. *Telektronikk,* 91(2), 130–138, 1995. (This issue.)

25 Svinnset, I. Aspects of dimensioning transmission resources in B-ISDN networks. *Telektronikk,* 91(2), 139–146, 1995. (This issue.)

26 Helvik, B E. Synthetic load generation for ATM traffic measurements. *Telektronikk,* 91(2), 174–194, 1995. (This issue.)

27 Heegaard, P E. Speed-up techniques for high-performance evaluation. *Telektronikk,* 91(2), 195–207, 1995. (This issue.)

A tribute to A.K. Erlang

BY ARNE MYSKJA

In this special *teletraffic* issue of the journal Telektronikk most attention naturally focuses on the state of the art problems. Those problems are as sophisticated and as diverse as the technologies and the applications themselves. Still the editor deems it highly appropriate to throw a glance back at the roots of the theories that have developed so impressively during close to ninety years. It all began in the days of transition from manual to automatic telephony, with extremely simple network structures, one type of service and a very limited number of users. Still, the basic concepts that were identified at the time are valid even today.

The Danish scientist and telephone laboratory manager Agner Krarup Erlang (1878 – 1929) has been called "the father of teletraffic theory". Among several works on statistics as well as electrotechnical matters, most of them related to his profession within the telephone business, one work published in 1917 is generally recognised as his most important contribution. That paper contains among other things the famous B-formula. The publication and the formula are explicitly mentioned in the original proposal of 1943 that later led to the CCIF decision of 1946 to adopt "Erlang"[1] as the unit of traffic intensity.

In 1948 the Danish Academy of Technical Sciences published in the Academy's Transaction series "The Life and Works of A. K. Erlang" [1] by authors E. Brockmeyer, H.L. Halstrøm and Arne Jensen. The publication was a highly laudable initiative, as the book contains a brief biography along with all the essential written works by Erlang, as well as elucidating articles extending on Erlang's theories in an updated mathematical form. Also, explanatory comments on each one of the included papers are offered. A second edition reprint appeared in 1960 in Acta Polytechnica Scandinavica.

Most of Erlang's publications were originally written in Danish and published in Danish journals of mathematics, physics or electrotechniques. Many were later published in foreign journals in English, French and/or German language. In [1] they all appear in English translation. Also a previously unpublished work is included.

Apart from a few mathematics studies and some mathematical tables, most of Erlang's works fall into two main groups: stochastic processes, with application to telephone traffic, and electrotechniques, mainly with application to signal transmission and measurements. Within both areas he utilises mathematics as the main tool. His works within the area of stochastic processes have had the most profound influence on later developments.

The first important publication appeared in *Nyt Tidsskrift for Matematik* in 1909, where he applies basic probability theory to the case of random calls, showing that the number of calls arriving within a given length of time is given by a Poisson distribution. Also, waiting time is treated in an initial way.

The reason for mentioning the 1909 paper in particular here is threefold: firstly, it indicates the beginning of the main work that Erlang carried out later; secondly, Erlang here (as also frequently later) relates his work to previous attempts at using probability theory as a tool in telephone traffic, with a special tribute to F. Johannsen; and thirdly, he makes the introductory statement that "... a special knowledge of telephonic problems is not at all necessary for the understanding [of the theory] ... ", this statement indicating the more general applicability of the theory.

In the eight years following his 1909 publication, Erlang seems to have been preoccupied with his assignment as a leader of the laboratory, as he mainly came out with electrotechnical work and presentation of numerical tables.

The already mentioned principal paper by Erlang , first published 1917 in Danish in *Elektroteknikeren,* later appeared in British, German and French journals. In ref. [1] the paper appears in English translation: *Solution of some Problems in the Theory of Probabilities of Significance in Automatic Telephone Exchanges*. In this paper the two main formulas connected with Erlang's name both appear, the blocking formula (B-formula) and the delay formula (D-formula).

Erlang's initial assumption in mathematical modelling of blocking and delay is the more difficult proposition of constant rather than exponential holding times. He arrives at the results that the assumption is of no consequence for the blocking probability, whereas for waiting time the exponential distribution means a great simplification, permitting the simple form of the D-formula for the waiting probability. For this case even the waiting time *distribution* in its general form is given without proof, along with mean waiting time. An initial discussion of gradings is also included, as a prelude to the later (1920) presentation of his interconnection formula.

Erlang applied with explicit mention the concept of "statistical equilibrium", and in later presentations he used the graphical tool of state transition diagrams. It is clear from Erlang's works that he was fully aware of the significance of distributions, with consequences far beyond mean values. In particular, he studied holding time distributions, with focus on exponential and constant time intervals. He treated the serial addition of exponential times, leading to the distribution that also carries his name (the Erlang-k distribution), and which is a general form, with constant and exponential distributions as limit cases.

In retrospect the scientific work of A. K. Erlang is quite impressive as a pioneering work, considering his understanding of the basic concepts, and his ability to formulate in concise mathematical terms the essential properties of telephone traffic in a way that opened the road of analysis and system dimensioning that have later proved so successful. Certainly, efforts in that direction had already been initiated by others, and simultaneous embryonic work was under way by a handful of other people. However, nobody challenges the unique position of A. K. Erlang on this arena.

In tribute to Erlang's pioneering work it is appropriate to include in the present teletraffic issue of Telektronikk a reprint from reference [1] of the 1917 publication.

Reference

1 Brockmeyer, E, Halstrøm, H L, Jensen, A. The life and works of A.K. Erlang. *Transactions of the Danish Academy of Mathematical Sciences,* 2, 1948.

1 *The use of upper and lower case (Erlang and erlang) is not consistent. In the primary reference [1] for the present article (p. 21) the quote from the CCIF decision uses "erlang", whereas in other context in the same reference "Erlang" is applied. Similarly, in the CCITT Blue Book, Fascicle I.3, the keyword is "erlang" (also all other keywords apply lower case), whereas in the definition of traffic unit "Erlang" is applied, with symbol E.*

Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges*

BY A K ERLANG

Summary. – Sections 1–7. First main problem: Systems without waiting arrangements. (Two different presuppositions.) Accompanied by Tables 1, 2, 3. Sections 8–9. Second main problem: systems with waiting arrangement (Two different presuppositions.) Accompanied by Tables 4, 5, 6, 7. Sections 10–12. Approximative methods, references, conclusion. Accompanied by Table 8.

1. First Main Problem. – Let us suppose that an automatic system is arranged in such a manner that there are provided x lines to take a certain number of subscribers. These x lines are said to be co-operative, or to constitute a "group" (or "team"). It is presupposed that all the lines disengaged are accessible. At present we will only speak of systems without waiting arrangements, i.e. systems in which the subscriber, when he finds that all x lines are engaged, replaces the receiver, and does not try to get connection again immediately. The probability of thus finding the lines engaged is called the loss, or degree of hindrance, and is here designated by B. With respect to the length of the conversations (sometimes called the holding-time), we will (for the present) suppose that it is constant, and it will be convenient to consider this quantity equal to 1 ("the natural time-unit"). With respect to the subscribers' calls, it is assumed that they are distributed quite accidentally throughout the time in question (e.g. that part of the day when the heaviest traffic usually occurs). This presupposition does not only imply that there must not be points of time within the period of time in consideration at which it may be expected in advance that there will be exceptionally many or few calls, but also that the calls must be mutually independent. In practice these presuppositions will, with great approximation, be fulfilled. The average number of calls per time-unit (intensity of traffic) is called y. The ratio of y to x, i.e. the traffic intensity per line, is designed by α; it is often called the efficiency of the group. We have to determine B (as a function of y and x). The exact expression for this is as follows:

$$B = \frac{\frac{y^x}{x!}}{1 + \frac{y}{1!} + \frac{y^2}{2!} + \ldots + \frac{y^x}{x!}} \qquad (1)$$

as proved in the following sections (2–5).

2. The following proof may be characterised as belonging to the mathematical statistics, and is founded on the theory of "statistical equilibrium" – a conception which is of great value in solving certain classes of problems in the theory of probabilities. Let us consider a very great number of simultaneously operating groups of lines of the previously described kind (number of lines x, traffic intensity y). If we examine a separate group at a definite moment, we may describe its momentary condition by stating, firstly, how many of the x lines (0, 1, 2, ... x) are engaged; and secondly, how much there is left of each of the conversations in question. If we examine the same group a short time dt later, we will find that certain changes of two different kinds have taken place. On the one hand, the conversations which were nearly finished will now be over, and the others have become a little older. On the other hand, new calls may have been made, which, however, will have significance only if not all the lines are engaged. (The probability of a new call during the short time dt is ydt.) We assume that we examine in this manner not only one group, but a very great number of groups, both with respect to the momentary condition and the manner in which this alters. The state, of which we thus can get an accurate description, if we use a sufficiently large material, has the characteristic property that, notwithstanding the aforesaid individual alterations, it maintains itself, and, when once begun, remains unaltered, since the alterations of the different kinds balance each other. This property is called "statistic equilibrium".

3. Temporarily as a postulate, we will now set forth the following description of the state of statistical equilibrium.

The probabilities that 0, 1, 2, 3, ... x lines are engaged are respectively –

$$S_0 = \frac{1}{1 + \frac{y}{1!} + \frac{y^2}{2!} + \ldots + \frac{y^x}{x!}}$$

$$S_1 = \frac{\frac{y}{1}}{1 + \frac{y}{1!} + \frac{y^2}{2!} + \ldots + \frac{y^x}{x!}}$$

$$S_2 = \frac{\frac{y^2}{2}}{1 + \frac{y}{1!} + \frac{y^2}{2!} + \ldots + \frac{y^x}{x!}} \qquad (2)$$

....

....

$$S_x = \frac{\frac{y^x}{x!}}{1 + \frac{y}{1!} + \frac{y^2}{2!} + \ldots + \frac{y^x}{x!}}$$

where the sum of all the probabilities is 1, as it should be. And we further postulate for each of the $x + 1$ aforesaid special conditions, that the still remaining parts of the current conversations ("remainders") will vary quite accidentally between the limits 0 and 1, so that no special value or combination of values is more probable than the others.

4. We shall prove that the thus described general state is in statistical equilibrium. For that purpose we must keep account of the fluctuations (increase and decrease), during the time dt, for the $x + 1$ different states, beginning with the first two. The transition from the first state S_0 to the second state S_1 amounts to

$$S_0 ydt,$$

while the transition from the second S_1 to the first S_0 amounts to

$$S_1 \cdot dt.$$

These quantities are according to (3) equal and thus cancel each other. Furthermore, the amount of transition from S_1 to S_2 is:

$$S_1 \cdot ydt,$$

* *First published in "Elektroteknikeren" vol. 13 (1917) p.5.*

Table 1.
Values of the Loss, or Grade of Service, B. (Formula (1), Section 1).

x	*α*	*y*	*B*
1	0.1	0.1	0.091
1	0.2	0.2	0.167
2	0.1	0.2	0.016
2	0.2	0.4	0.054
2	0.3	0.6	0.101
3	0.1	0.3	0.003
3	0.2	0.6	0.020
3	0.3	0.9	0.050
3	0.4	1.2	0.090
4	0.1	0.4	0.001
4	0.2	0.8	0.008
4	0.3	1.2	0.026
4	0.4	1.6	0.056
5	0.2	1.0	0.003
5	0.3	1.5	0.014
5	0.4	2.0	0.037
5	0.5	2.5	0.070
6	0.2	1.2	0.001
6	0.3	1.8	0.008
6	0.4	2.4	0.024
6	0.5	3.0	0.052
8	0.3	2.4	0.002
8	0.4	3.2	0.011
8	0.5	4.0	0.030
10	0.3	3	0.001
10	0.4	4	0.005
10	0.5	5	0.018
10	0.6	6	0.043
10	0.7	7	0.079
20	0.4	8	0.000
20	0.5	10	0.002
20	0.6	12	0.010
20	0.7	14	0.079
30	0.5	15	0.000
30	0.6	18	0.003
30	0.7	21	0.014
40	0.5	20	0.000
40	0.6	24	0.001
40	0.7	28	0.007

Table 2.
Values of the intensity of traffic, y, as a function of the number of lines, x, for a loss of 1, 2, 3, 4 ‰.

x	1 ‰	2 ‰	3 ‰	4 ‰
1	0.001	0.002	0.003	0.004
2	0.046	0.065	0.081	0.094
3	0.19	0.25	0.29	0.32
4	0.44	0.53	0.60	0.66
5	0.76	0.90	0.99	1.07
6	1.15	1.33	1.45	1.54
7	1.58	1.80	1.95	2.06
8	2.05	2.31	2.48	2.62
9	2.56	2.85	3.05	3.21
10	3.09	3.43	3.65	3.82
11	3.65	4.02	4.26	4.45
12	4.23	4.64	4.90	5.11
13	4.83	5.27	5.56	5.78
14	5.45	5.92	6.23	6.47
15	6.08	6.58	6.91	7.17
16	6.72	7.26	7.61	7.88
17	6.38	7.95	8.32	8.60
18	8.05	8.64	9.03	9.33
19	8.72	9.35	9.76	10.07
20	9.41	10.07	10.50	10.82
25	12.97	13.76	14.28	14.67
30	16.68	17.61	18.20	18.66
35	20.52	21.56	22.23	22.75
40	24.44	25.6	26.3	26.9
45	28.45	29.7	30.5	31.1
50	32.5	33.9	34.8	35.4
55	36.6	38.1	39.0	39.8
60	40.8	42.3	43.4	44.1
65	45.0	46.6	47.7	48.5
70	49.2	51.0	52.1	53.0
75	53.5	55.3	56.5	57.4
80	57.8	59.7	61.0	61.9
85	62.1	64.1	65.4	66.4
90	66.5	68.6	69.9	70.9
95	70.8	73.0	74.4	75.4
100	75.2	77.5	78.9	80.0
105	79.6	82.0	83.4	84.6
110	84.1	86.4	88.0	89.2
115	88.5	91.0	92.5	93.7
120	93.0	95.5	97.1	98.4

and, conversely, the transition from S_2 to S_1 is:

$$S_2 \cdot 2 \cdot dt,$$

which two quantities also are equal and cancel each other.

Finally, we have

$$S_{x\text{-}1} \cdot y \cdot dt$$

and

$$S_x \cdot x \cdot dt,$$

which also cancel each other. The result is that the reciprocal changes which take place between the $x + 1$ different states during the time dt, compensate each other, so that the distribution remains unaltered. We still have to prove that neither will there be any alterations in the distribution of the magnitude of the remainders, i.e. that the decrease and increase, also in this respect, compensate each other.

5. Let us consider the cases in which the number of current conversations is n, and among these cases, more especially those in which the magnitudes of the n remain-

Table 3 a.
The "Loss" (in ‰ by 3 different arrangements (one with "Grading and Interconnecting").

y	3	4	5	6	7	8	9	10	11	12
1) x = 10, with 10 contacts	0.8	5.3	18.4	43.1	–	–	–	–	–	–
1) x = 18, with 18 contacts	–	–	–	–	0.2	0.9	2.9	7.1	14.8	26.5
3) x = 18, with 10 contacts	–	–	–	–	1.1	3.1	7.4	15.1	26.8	42.8

Table 3 b.
Values of a and y by different arrangements for a loss of 1 ‰.

	α	y
x = 10; 10 contacts	0.31	3.1
x = 18; 10 -	0.38	6.9
x = ∞; 10 -	0.50	–

ders lie, respectively, between the following limits:

t_1 and $t_1 + \Delta_1$,
t_2 and $t_2 + \Delta_2$,
.. ..
.. ..
t_n and $t_n + \Delta_n$.

The probability of this is (according to Section 3):

$$\Delta_1 \cdot \Delta_2 \cdot \Delta_3 \ldots \Delta_n \cdot S_n.$$

During the time *dt* there may occur, in four different ways, both increase and decrease.

Firstly, transition to S_{n+1}; namely, if a call arrives; the probability of this will be:

$$\Delta_1 \cdot \Delta_2 \cdot \Delta_3 \ldots \Delta_n \cdot S_n \cdot y \cdot dt.$$

Secondly, transition from S_{n+1}; namely, if one among the $n + 1$ current conversations finishes during the time *dt*, and, thereafter, the *n* remainders lie between the above settled limits. The corresponding probability is:

$$\Delta_1 \cdot \Delta_2 \cdot \Delta_3 \ldots \Delta_n\,(n+1)\,S_{n+1} \cdot dt,$$

which is equal to the preceding one.

Thirdly, transition from S_n itself; namely, if, among the *n* remainders, the $n - 1$ lie between the settled limits, and the one lies just below the lower limit in question, at a distance shorter than *dt*. The probability for this will be:

$$\Delta_1 \cdot \Delta_2 \cdot \Delta_3 \ldots \Delta_n \left(\frac{1}{\Delta_1} + \frac{1}{\Delta_2} + \ldots + \frac{1}{\Delta_n}\right) S_n \cdot dt$$

Fourthly, transition to S_n itself; namely, if, among the *n* remainders, the $n - 1$ lie between the settled limits, and the one lies just below the upper limit, at a distance shorter than *dt*. The probability of this eventuality is obviously equal to the preceding one.

Thus, there is a balance. So it is proved by this that there will be statistical equilibrium. On the other hand, any other supposition than the one set forth in Section 3 will at once be seen to be inconsistent with statistic equilibrium. The formulae in Section 3 are now proved, and thereby the proposition in Section 1 is also proved.

6. The above presupposition, that all conversations are of equal length, applies with great approximation to trunk-line conversations, but not, of course, to the usual local conversations. Now, a statistic investigation, which I have undertaken, shows that the duration of these conversations is ruled by a simple law of distribution, which may be expressed as follows:

The probability that the duration will exceed a certain time *n* is equal to

$$e^{-n},$$

when the average duration is taken to be equal to 1, as before. Or, in other words, the probability that a conversation which has been proceeding for some time is nearly finished, is quite independent of the length of the time which has already elapsed. The average number of conversations finished during the time *dt* (per current conversation) will be equal to *dt.* It is now easy to see that we must arrive at the same expression (1) for *B* as under the former presupposition, only that the proof becomes somewhat simpler, because it is necessary to take into account only the number of current conversations without paying any attention to their age. (It will appear from the following that the two aforesaid presuppositions do not lead to the same result in *all* problems.)

7. In Table 1 are shown some numerical values of the "loss" *B* as dependent of *x* and *y* (or α), and as given by the proposed theory.

In Table 2 the results of formula (1) are presented in another form, which is probably the one that is most useful in practice; *x* and *B* are here entry numbers, and the table gives *y* as a function of *x* and *B*.

In Table 3a only the first and second lines threat of systems with "pure" groups (to which formula (1) applies). The values given in the third line correspond to a different system, in which a special arrangement, the so-called "grading and interconnecting", is used. We may describe this arrangement as follows:

The number of contacts of the selectors (here ten) is less than the number of lines (here eighteen) in the "group". Thus each call searches not all eighteen but only ten lines. It is hereby presupposed (for the sake of simplicity) that the ten lines are each time accidentally chosen, out of the eighteen, and that they are tested one after the other according to an arbitrary selection. The method of calculation here to be used may be considered as a natural extension of the method which leads to formula (1), but it is, of course, a little more complicated. A few results of this kind of calculating are given in the two Tables 3a and 3b. Finally, I want to point out that the systems for "grading and interconnecting" being used in practice at present, which I, however, do not know in detail, are said to deviate a little from the description given here, and, therefore, it may be expected that they will give somewhat less favourable results.

8. Second Main Problem. – The problem to be considered now concerns systems

Table 4. (x = 1).

a \ n	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.00	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.05	0.950	0.955	0.960	0.964	0.969	0.974	0.979	0.984	0.989	0.994	0.999
0.10	0.900	0.909	0.918	0.927	0.937	0.946	0.956	0.965	0.975	0.985	0.995
0.15	0.850	0.863	0.876	0.889	0.903	0.916	0.930	0.944	0.958	0.973	0.988
0.20	0.800	0.816	0.833	0.849	0.867	0.884	0.902	0.920	0.939	0.958	0.977
0.25	0.750	0.769	0.788	0.808	0.829	0.850	0.871	0.893	0.916	0.939	0.963
0.30	0.700	0.721	0.743	0.766	0.789	0.814	0.838	0.864	0.890	0.916	0.946
0.35	0.650	0.673	0.697	0.722	0.748	0.774	0.802	0.830	0.860	0.891	0.922
0.40	0.600	0.624	0.650	0.677	0.704	0.733	0.763	0.794	0.826	0.860	0.895
0.45	0.550	0.575	0.602	0.630	0.658	0.689	0.720	0.754	0.788	0.825	0.863
0.50	0.500	0.526	0.553	0.581	0.611	0.642	0.675	0.710	0.746	0.784	0.824
0.55	0.450	0.475	0.502	0.531	0.561	0.592	0.626	0.661	0.699	0.738	0.780
0.60	0.400	0.425	0.451	0.479	0.508	0.540	0.573	0.609	0.646	0.686	0.729
0.65	0.350	0.374	0.399	0.425	0.454	0.484	0.517	0.552	0.589	0.628	0.670
0.70	0.300	0.322	0.345	0.370	0.397	0.426	0.457	0.490	0.525	0.563	0.604
0.75	0.250	0.269	0.290	0.313	0.337	0.364	0.392	0.423	0.456	0.491	0.529
0.80	0.200	0.217	0.235	0.254	0.275	0.298	0.323	0.350	0.379	0.411	0.445
0.85	0.150	0.163	0.178	0.194	0.211	0.339	0.250	0.272	0.296	0.322	0.351
0.90	0.100	0.109	0.120	0.131	0.143	0.157	0.172	0.188	0.205	0.225	0.246
0.95	0.050	0.055	0.060	0.066	0.073	0.080	0.088	0.097	0.107	0.118	0.129
1.00	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 5. (x = 2).

a \ n	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.00	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.05	0.995	0.996	0.997	0.998	0.998	0.999	0.999	1.000	1.000	1.000	1.000
0.10	0.982	0.985	0.988	0.991	0.993	0.995	0.997	0.998	0.999	1.000	1.000
0.15	0.962	0.968	0.974	0.980	0.985	0.989	0.993	0.996	0.998	0.999	1.000
0.20	0.936	0.946	0.956	0.964	0.973	0.980	0.987	0.992	0.995	0.998	0.999
0.25	0.903	0.918	0.933	0.946	0.958	0.969	0.978	0.986	0.992	0.996	0.998
0.30	0.866	0.886	0.905	0.922	0.939	0.953	0.967	0.978	0.986	0.992	0.995
0.35	0.825	0.849	0.872	0.895	0.915	0.935	0.952	0.966	0.978	0.987	0.991
0.40	0.779	0.808	0.835	0.862	0.888	0.911	0.933	0.952	0.967	0.978	0.985
0.45	0.730	0.762	0.794	0.825	0.855	0.883	0.909	0.932	0.951	0.966	0.975
0.50	0.677	0.712	0.748	0.783	0.817	0.849	0.880	0.907	0.931	0.949	0.961
0.55	0.621	0.658	0.697	0.735	0.773	0.809	0.844	0.875	0.903	0.925	0.941
0.60	0.561	0.601	0.641	0.681	0.722	0.762	0.800	0.836	0.868	0.894	0.913
0.65	0.499	0.539	0.580	0.622	0.664	0.706	0.748	0.787	0.822	0.852	0.875
0.70	0.435	0.473	0.514	0.556	0.599	0.642	0.685	0.726	0.764	0.798	0.824
0.75	0.368	0.404	0.442	0.483	0.525	0.568	0.611	0.653	0.692	0.728	0.757
0.80	0.298	0.331	0.366	0.403	0.442	0.482	0.523	0.564	0.603	0.639	0.669
0.85	0.227	0.254	0.283	0.315	0.348	0.384	0.420	0.457	0.493	0.527	0.556
0.90	0.153	0.173	0.195	0.219	0.244	0.272	0.300	0.330	0.359	0.387	0.412
0.95	0.077	0.088	0.101	0.114	0.129	0.144	0.161	0.179	0.196	0.214	0.230
1.00	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 6. (x = 3).

a \ n	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.00	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.05	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
0.10	0.996	0.997	0.998	0.999	0.999	1.000	1.000	1.000	1.000	1.000	1.000
0.15	0.989	0.992	0.994	0.996	0.997	0.998	0.999	1.000	1.000	1.000	1.000
0.20	0.976	0.982	0.987	0.991	0.994	0.996	0.998	0.999	1.000	1.000	1.000
0.25	0.958	0.967	0.975	0.983	0.988	0.993	0.996	0.998	0.999	1.000	1.000
0.30	0.933	0.948	0.960	0.971	0.980	0.987	0.992	0.996	0.998	0.999	0.999
0.35	0.903	0.923	0.940	0.956	0.969	0.979	0.987	0.993	0.996	0.998	0.999
0.40	0.866	0.892	0.915	0.936	0.953	0.968	0.980	0.988	0.993	0.996	0.998
0.45	0.823	0.855	0.884	0.910	0.934	0.953	0.969	0.980	0.988	0.993	0.995
0.50	0.775	0.812	0.847	0.879	0.908	0.933	0.953	0.969	0.980	0.987	0.991
0.55	0.720	0.762	0.803	0.841	0.876	0.906	0.932	0.952	0.967	0.977	0.983
0.60	0.660	0.706	0.752	0.795	0.835	0.872	0.903	0.929	0.948	0.962	0.971
0.65	0.595	0.644	0.693	0.740	0.786	0.827	0.864	0.895	0.920	0.938	0.951
0.70	0.524	0.574	0.625	0.676	0.725	0.771	0.813	0.849	0.879	0.902	0.919
0.75	0.448	0.496	0.548	0.600	0.651	0.700	0.746	0.787	0.821	0.849	0.871
0.80	0.367	0.413	0.461	0.511	0.562	0.611	0.659	0.702	0.740	0.773	0.799
0.85	0.282	0.322	0.364	0.409	0.455	0.501	0.547	0.590	0.629	0.663	0.693
0.90	0.192	0.222	0.255	0.291	0.328	0.366	0.405	0.442	0.477	0.509	0.538
0.95	0.098	0.115	0.134	0.155	0.177	0.201	0.225	0.249	0.273	0.295	0.316
1.00	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

with waiting arrangements. Here, the problem to be solved is determining the probability S (> n) of a waiting time greater than an arbitrary number n, greater than or equal to zero. The last case is the one which is most frequently asked for. In the same manner we define S (< n) where S (< n) + S (> n) = 1. Furthermore, we may ask about the average waiting time M. We shall answer these questions in the following. Here, too, we may begin by assuming that the duration of the conversations is constant and equal to 1. The accurate treatment of this case gives rise to rather difficult calculations, which, however, are unavoidable. Among other things, we find that we cannot use the same formula for S (> n) for all values of n, but we must distinguish between the various successive "periods", or spaces of time of the length l. In practice, however, the first period will, as a rule, be the most important. I shall content myself by giving, without proof, the necessary formulae for the cases of x = 1, 2, and 3, and then (chiefly for the purpose of showing the possibility of carrying out the practical calculation) the corresponding numerical results, also for x = 1, 2, 3. Formulae and results for x = 1 have already been published in an article in "Nyt Tidsskrift for Mathematik", B, 20, 1909. The formulae for greater values of x, e.g. x = 10, x = 20 are quite analogous to those given here.

Collection of formulae

Presupposition: the duration of conversations is constant and equal to l

Denotations:

x is the number of co-operating lines

y is the intensity of traffic (average number of calls during unit of time)

$$\frac{y}{x} = \alpha$$

S (> n) is the probability of a waiting time greater than n

S (< n) is the probability of a waiting time less than, or equal to n

$ny = z$

$z - y = u$

$z - 2y = v$, et cetera.

M = the average waiting time.

I. Formulae for the case of x = 1:

a) First period, $0 < n < 1$:
$S(<n) = a_0 \cdot e^z$,

where $a_0 = 1 - a$

b) Second period, $1 < n < 2$:
$S(<n) = (b_1 - b_0 u)e^u$,

where $b_1 = a_0 e^y$
$b_0 = a_0$

c) Third period, $2 < n < 3$:
$S(<n) = (c_2 - c_1 v + {}^1\!/_2\, c_0 v^2)e^v$

where $c_2 = (b_1 - b_0 y)e^y$
$c_1 = b_1$
$c_0 = b_0$

et cetera.

$$M = \frac{1}{y}\left((1-b_1) + (1-c_2) + (1-d_3) + \ldots\right)$$

$$= \frac{1}{2} \cdot \frac{\alpha}{1-\alpha}$$

II. Formulae for the case of x = 2:

a) First period, $0 < n < 1$:
$S(<n) = (a_1 - a_0 z)e^z$

where

$$a_1 = 2(1-\alpha)\frac{\alpha}{\alpha - \beta},$$

$$a_0 = -2(1-\alpha)\frac{\beta}{\alpha - \beta}$$

β denoting the negative root of the equation

$$\beta e^{-\beta} = -\alpha e^{-\alpha}.$$

b) Second period, $1 < n < 2$:

$S(< n) = (b_3 - b_2 u + {}^1/_2\, b_1 u^2 - {}^1/_6\, b_0 u^3)e^u$

where $b_3 = (a_1 - a_0 y)e^y$
$b^2 = a_0 e^y$
$b_1 = a_1$
$b_0 = a_0$

c) Third period, $2 < n < 3$:

$S(< n) = (c_5 - c_4 v + {}^1/_2\, c_3 v^2 - {}^1/_6\, c_2 v^3 + {}^1/_{24}\, c_1 v^4 - {}^1/_{120}\, c_0 v^5)e^v$

where $c_5 = (b_3 - b_2 y + {}^1/_2\, b_1 y^2 - {}^1/_6\, b_0 y^3)e^y$
$c_4 = (b_2 - b_1 y + {}^1/_2\, b_0 y^2)e^y$
$c_3 = b_3$
$c_2 = b_2$
$c_1 = b_1$
$c_0 = b_0$

et cetera.

$$M = \frac{1}{y}\,((1 - b_2) + (1 - b_3) + (1 - c_4)$$
$$+(1 - c_5) + (1 - d_6) + (1 - d_7) + \ldots)$$

III. Formulae for the case of $x = 3$

a) First period, $0 < n < 1$:

$S(< n) = (a_2 - a_1 z + {}^1/_2\, a_0 z^2)e^z$

where

$$a_2 = 3(1-\alpha)\frac{\alpha^2}{(\alpha-\beta)(\alpha-\gamma)}$$

$$a_1 = -3(1-\alpha)\frac{\alpha(\beta+\gamma)}{(\alpha-\beta)(\alpha-\gamma)}$$

$$a_0 = 3(1-\alpha)\frac{\beta\cdot\gamma}{(\alpha-\beta)(\alpha-\gamma)}$$

as $\beta \cdot e^{-\beta} = \alpha \cdot e^{-\alpha} \cdot k$
$\gamma \cdot e^{-\gamma} = \alpha \cdot e^{-\alpha} \cdot k^2$

We understand by k a complex value of $\sqrt[3]{1}$

b) Second period, $1 < n < 2$:

$S(< n) = (b_5 - b_4 u + {}^1/_2\, b_3 u^2 - {}^1/_6\, b_2 u^3 + {}^1/_{24}\, b_1 u^4 - {}^1/_{120}\, b_0 u^5)e^u$,

where $b_5 = (a_2 - a_1 y + {}^1/_2\, a_0 y^2)e^y$
$b_4 = (a_1 - a_0 y)e^y$
$b_3 = a_0 e^y$
$b_2 = a_2$
$b_1 = a_1$
$b_0 = a_0$

c) Third period, $2 < n < 3$:

$S(< n) = (c_8 - c_7 v + {}^1/_2\, c_6 v^2 - {}^1/_6\, c_5 v^3 + {}^1/_{24}\, c_4 v^4 - {}^1/_{120}\, c_3 v^5 + {}^1/_{720}\, c_2 v^6 - {}^1/_{5040}\, c_1 v^7 + {}^1/_{40320}\, c_0 v^8)e^v$,

where $c_8 = (b_5 - b_4 y + {}^1/_2\, b_3 y^2 - {}^1/_6\, b_2 y^3 + {}^1/_{24}\, b_1 y^4 - {}^1/_{120}\, b_0 y^5)e^y$

$c_7 = (b_4 - b_3 y + {}^1/_2\, b_2 y^2 - {}^1/_6\, b_1 y^3 + {}^1/_{24}\, b_0 y^4)e^y$

$c_6 = (b_3 - b_2 y + {}^1/_2\, b_1 y^2 - {}^1/_6\, b_0 y^3)e^y$
$c_5 = b_5$
$c_4 = b_4$
$c_3 = b_3$
$c_2 = b_2$
$c_1 = b_1$
$c_0 = b_0$,

et cetera.

$$M = \frac{1}{7}\,((1 - b_3) + (1 - b_4) + (1 - b_5)$$
$$+(1 - c_6) + (1 - c_7) + (1 - c_8) + \ldots)$$

9. There still remains the problem of investigating the magnitude of the waiting times in systems with waiting arrangement under the second presupposition, namely, that the durations of the conversations vary in the manner already described in Section 6.

Here we find, without difficulty, the following two formulae:

$$S(> 0) = c \qquad (3)$$
$$S(> n) = c \cdot e^{-(x-y)n} \qquad (4)$$

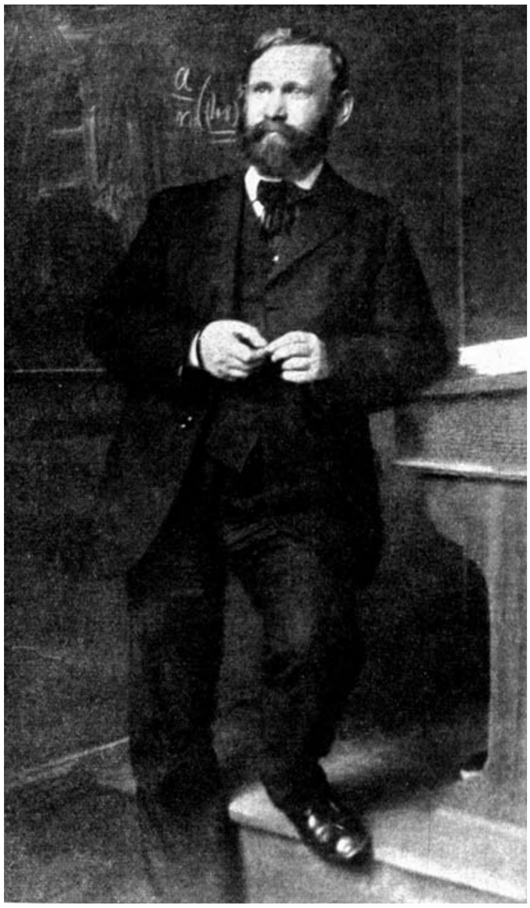
where

$$c = \frac{\frac{y^x}{x!}\cdot\frac{x}{x-y}}{1 + \frac{y}{1!} + \frac{y^2}{2!} + \ldots + \frac{y^{x-1}}{(x-1)!} + \frac{y^x}{x!}\cdot\frac{x}{x-y}} \qquad (5)$$

while x and y have the same significance as before, and the average duration of a conversation is equal to 1. The formula is exact for all values of $n > 0$.

For the average waiting time we get the formula:

$$M = \frac{c}{x-y} \qquad (6)$$



A.K. Erlang in 1910, 32 years old (from [1])

The numerical calculation causes no special difficulty. It ought, perhaps, to be pointed out that, both here and in Section 8, it is presupposed that the waiting calls are despatched in the order in which they have been received. If this does not take place in practice, it will, of course, have a slight effect upon the value of $S (> n)$, but not at all on the value of M, neither on $S (> 0)$.

10. Approximate Formulae. – The exact formulae given above are throughout so convenient, that there is hardly any need of approximative formulae. This does not, however, apply to the formulae which concern the second main problem, first presupposition. Therefore, it may be worth while to mention a couple of approximative methods which quickly lead to a serviceable result, at least in such cases as have the greatest practical significance.

One of these methods has already been used by me, at the request of Mr. *P.V. Christensen,* Assistant Chief Engineer to the Copenhagen Telephone Company, for calculating the explicit tables given in the first pages of his fundamental work, “The Number of Selectors in Automatic

Telephone Systems" (published in the Post Office Electrical Engineers' Journal, October, 1914, p. 271; also in "Elektroteknikeren", 1913, p. 207; "E.T.Z.", 1913, p. 1314).

Since the method used has not been described in full, I shall here say a few words about the same. The probability of just x calls being originated during a period of time for which the average number is y, is, as well known, under the usual presuppositions (Section 1):

$$e^{-y}\frac{y^x}{x!}$$

The mathematical theorem here used is due to *S.D. Poisson* ("Recherches sur la probabilité, etc.", 1835), and has later been studied by *L.v. Bortkewitsch* ("Das Gesetz der kleinen Zahlen", 1898). The function has been tabulated by the latter (*loc. cit.*), and later by *H.E. Soper* ("Biometrica", vol. X, 1914; also in *K. Pearson* "Tables for Statisticians, etc.", 1914).

Thus the probability of x or more calls during the mentioned period of time is:

$$P = e^{-y}\frac{y^x}{x!} + e^{-y}\frac{y^{x+1}}{(x+1)!} + e^{-y}\frac{y^{x+2}}{(x+2)!} + ... \quad (7)$$

It will then be seen that P, in many cases, viz. when y is not unproportionally great, will be a good approximate value for the fraction of the calls which will find all the lines engaged (or for "the probability of not getting through"). Thus P in the case of exchanges without waiting arrangements approximates the *"loss"*, and here gives obviously a somewhat too great value. In exchanges with waiting arrangement P approximates the quantity $S\ (> 0)$, the probability of delay, and gives here a somewhat too small value. Or, if it is the fraction named above which is given beforehand, as is generally the case in practice, where often the value 0.001 is used, the formula will show the connexion between y and x. The values of y found in this manner (see Table 8) will never deviate 5 per cent. from the correct values in systems without waiting arrangement; never 1 per cent. in systems with waiting arrangement (both presuppositions), if we take the named, frequently used value of P = 0.001. Possible intermediate systems between the two main classes of exchanges may, of course, be treated with good results according to the same method.

Table 7.
Systems with Waiting Arrangement (Second Presupposition). Values of S (> n) and M.

x	*α*	y	*S*(> 0)	*S*(> 0.1)	*S*(> 0.2)	*M*
1	0.1	0.1	0.100	0.091	0.084	0.111
1	0.2	0.2	0.200	0.185	0.170	0.250
2	0.1	0.2	0.018	0.015	0.013	0.010
2	0.2	0.4	0.067	0.057	0.049	0.042
2	0.3	0.6	0.138	0.120	0.104	0.099
3	0.1	0.3	0.004	0.003	0.002	0.001
3	0.2	0.6	0.024	0.019	0.015	0.010
3	0.3	0.9	0.070	0.057	0.046	0.033
3	0.4	1.2	0.141	0.118	0.099	0.078
4	0.1	0.4	0.001	0.001	0.000	0.000
4	0.2	0.8	0.010	0.007	0.005	0.003
4	0.3	1.2	0.037	0.028	0.022	0.013
4	0.4	1.6	0.091	0.072	0.056	0.038
5	0.2	1.0	0.004	0.003	0.002	0.001
5	0.3	1.5	0.020	0.014	0.010	0.006
5	0.4	2.0	0.060	0.044	0.033	0.020
5	0.5	2.5	0.130	0.102	0.079	0.052
6	0.2	1.2	0.002	0.001	0.001	0.000
6	0.3	1.8	0.011	0.007	0.005	0.003
6	0.4	2.4	0.040	0.026	0.018	0.011
6	0.5	3.0	0.099	0.073	0.054	0.033
8	0.3	2.4	0.004	0.002	0.001	0.001
8	0.4	3.2	0.018	0.011	0.007	0.004
8	0.5	4.0	0.059	0.040	0.026	0.015
10	0.3	3	0.001	0.001	0.000	0.000
10	0.4	4	0.009	0.005	0.003	0.001
10	0.5	5	0.036	0.022	0.013	0.007
10	0.6	6	0.102	0.068	0.046	0.026
10	0.7	7	0.222	0.165	0.122	0.074
20	0.4	8	0.000	0.000	0.000	0.000
20	0.5	10	0.008	0.001	0.001	0.000
20	0.6	12	0.024	0.011	0.005	0.003
20	0.7	14	0.094	0.052	0.028	0.016
22	0.5	11.0	0.002	0.001	0.000	0.000
22	0.6	13.2	0.018	0.007	0.003	0.002
22	0.7	15.4	0.081	0.042	0.022	0.012
30	0.5	15	0.000	0.000	0.000	0.000
30	0.6	18	0.007	0.002	0.001	0.001
30	0.7	21	0.044	0.018	0.007	0.005
40	0.5	20	0.000	0.000	0.000	0.000
40	0.6	24	0.002	0.000	0.000	0.000
40	0.7	28	0.022	0.007	0.002	0.002

If, in systems with waiting arrangement, we ask about the number of waiting times beyond a certain limit *n*, $S\ (> n)$, an extension of the same formula may be used, y being replaced by $y(1 - n)$. The method is best suited for small values of n, and the error goes to the same side as mentioned above. Furthermore, it may be mentioned in this connexion that if we use, in the case considered, the formulae following from presupposition No. 2, instead of those based upon presupposition No. 1, the errors thus introduced will be small, as a rule; they go, this time, in such a direction that we get too *great* values for $S\ (> 0)$ and $S\ (> n)$; or, if it is y which is sought, there will be too small values for y.

Table 8.
Values of y as a function of x, for P = 0.001 – 0.002 – 0.003 – 0.004.

x	1 ‰	2 ‰	3 ‰	4 ‰
1	0.001	0.002	0.003	0.004
2	0.045	0.065	0.08	0.09
3	0.19	0.24	0.28	0.31
4	0.42	0.52	0.58	0.63
5	0.73	0.86	0.95	1.02
6	1.11	1.27	1.38	1.46
7	1.52	1.72	1.85	1.95
8	1.97	2.20	2.35	2.47
9	2.45	2.72	2.89	3.02
10	2.96	3.25	3.45	3.60
11	3.49	3.82	4.03	4.19
12	4.04	4.41	4.62	4.81
13	4.61	5.00	5.24	5.43
14	5.19	5.61	5.87	6.07
15	5.79	6.23	6.51	6.72
16	6.40	6.86	7.16	7.38
17	7.03	7.51	7.82	8.06
18	7.66	8.17	8.49	8.74
19	8.31	8.84	9.18	9.44
20	8.96	9.51	9.87	10.14
21	9.61	10.20	10.57	10.84
22	10.28	10.89	11.27	11.56
23	10.96	11.59	11.98	12.28
24	11.65	12.29	12.70	13.01
25	12.34	13.00	13.42	13.74
30	15.87	16.6	17.1	17.4
35	19.5	20.4	20.9	21.3
40	23.5	24.2	24.8	25.2
45	27.1	28.1	28.7	29.2
50	30.9	32.0	32.7	33.2
55	34.9	36.0	36.8	37.3
60	38.9	40.1	40.9	41.4
65	43.0	44.2	45.0	45.6
70	47.0	48.3	49.2	49.8
75	51.0	52.4	53.3	54.0
80	55.1	56.6	57.6	58.3
85	59.3	60.9	61.8	62.5
90	63.5	65.1	66.1	66.9
95	67.7	69.3	70.4	71.1
100	71.9	73.6	74.7	75.5
105	76.2	77.9	79.0	79.8
110	80.4	82.2	83.3	84.2
115	84.7	86.6	87.7	88.5
120	89.0	90.9	92.1	93.0

11. It will be too lengthy to describe or mention, in this connexion, all the systematic practical experiments and measurements (also only partly published), which of late years have been made, partly by the firms in question (especially, Siemens and Halske, and Western Electric Co.), partly by others, or such purely empirical formulae as have thus been set forth. On the other hand, it would be incorrect to neglect one or two interesting theoretical works from recent years, which directly concern one of the problems treated above. In his doctor's thesis, Mr. *F. Spiecker* ("Die Abhängigkeit des erfolgreichen Fernsprechanrufes von der Anzahl der Verbindungsorgane", 1913), has indicated a method for determining the loss in systems without waiting arrangement, which (as he himself admits) is not quite free from errors, and which, besides, is so complicated that it can hardly find application in practice. It should be emphasized, however, that the results in the cases in which the author has completed his calculations, lie very near the results of formula (1) given above. In the same work is also given an approximative formula, which can best be compared with the formula for *P* (Section 10 above). The difference is exclusively due to a somewhat deviating, and probably less practical, formulation of the problem. Mr. *W.H. Grinsted*, in his treatise, "A Study of Telephone Traffic Problems, etc." (Post Office Electrical Engineers' Journal, April 1915), presents a solution of the same problem. Since this solution has, probably, by many readers as well as by the author himself, been considered mathematically exact, it should be noticed that an error has occurred in the derivation of the formula in question and that, for this reason, the formula gives rather incorrect results. It should be added that the treatise is a reprint of an older work from 1907 (which I have not had opportunity to examine). In spite of the faulty results, Grinsted's work is, however, when its time of publication is considered, of no little merit.

12. In closing this article, I feel called upon to render my best thanks to Mr. *F. Johannsen*, Managing Director of the Copenhagen Telephone Co., not only for his interest in the investigations recorded here, but also for his energetic initiative in starting rational and scientific treatment of many different problems in connexion with telephone traffic. I also owe many thanks to Mr. *J.L.W.V. Jensen*, Engineer-in-Chief to the same Company, for his valuable assistance especially in the treatment of some mathematical difficulties.

The life and work of Conny Palm — some personal comments and experiences

BY ROLF B HAUGEN

This paper is an introduction to the life and works of the great Swedish teletraffic theoretician Conny Palm. The focus is on how I *personally* have used and experienced Palm's work throughout the years rather than trying to make an exhausted review. Through this flash back of my years of work in teletraffic, it is unavoidable not to mention a few other teletraffic researchers that came into "my way". Nevertheless, the theme throughout the paper is always Palm and his work.

1 Introduction

The science of *Teletraffic theory,* founded by Erlang in the beginning of this century, has always had a strong foothold in Scandinavia. The field was first dominated by the Danes, then gradually Swedish groups became dominant, until finally, on the threshold of the seventies, a relatively strong build-up started in Norway. By that time professor Arne Myskja and his colleagues in Trondheim were engaged in traffic measurements with subsequent analyses and several interesting studies of subscriber behavior. At Kjeller, Richard Solem was busy building up a scientific group at Norwegian Telecom Research (NTR) with a rather broad focus. The teletraffic group at NTR, the T-group, was soon to be a force centre in the Norwegian community, initiating activities in Trondheim as well as numerous places in Oslo, like the Norwegian Computer Centre, STK, EB, other departments of Norwegian Telecom, to mention but a few.

Personally, I had the privilege to join the Kjeller group in the middle of the seventies after some years as 'traveling' physicist. Coming from the University of Oslo rather than from NTH as did the others in the group, I was perhaps a bit of an outsider. However, since we were all physicists of one kind or another, I quickly fell in with the *professional* as well as the *social* activities in a fairly acceptable way. And the atmosphere and activities in the group were just incredible! It is rare in any profession to meet so many extraordinary gifted people in one and the same group. The T-group soon became a concept at the institute. Practical jokes were numerous and legendary, and in social activities like sports, games, partying, etc., the challenge at NTR was always: the T-group versus the rest! But the professional flag was nevertheless kept high!

Although my main fields of study were physics and mathematics, I embarked on a course in statistics, quite by impulse, at one point during my studies. By chance, I chose the subject *Probability theory* with a text book written by William Feller, a person unknown to me at that time. But seldom have I been more fascinated! It was obvious that Feller had practical experience in the field of probability; his examples and explanations were related to real life situations. And his approach and mathematical treatment of the problem was done in such an intuitive way that we hardly realized that we, in fact, were working with the fundamentals of the underlying theory. Even as a student I grasped the freshness of his treatment and was stricken by his pedagogic ability in explaining intricate problems in a simple way.

Feller brought me into the world of stochastic processes, a world fascinating in itself. But equally fascinating was Feller's treatment. I certainly remember being introduced to the state equations, the so called birth and death processes based on equilibrium considerations. Feller also introduced me to the *generating functions,* a powerful means of solving discrete processes. It was incredible with which elegance and virtuosity Feller solved intricate probability problems with the help of these functions! At least, so it seemed to me at that stage of my education.

Hardly could I then know that later, and for a long period of my life, I was to have the above mentioned processes and solution techniques as my daily work. I remember vaguely that I was introduced to the names Erlang and Palm; Erlang being the person behind the state equations that became the origin of queuing theory, and Palm one of the pioneers in establishing the mathematical fundament of those theories. I also seem to remember that some of the examples and solutions in the text book went back to Palm. Even if I at that time had no inclination towards telecommunications, I certainly grasped that Erlang had to do with telephony. Palm's background, though, was more diffuse to me.

Another eight years were to pass before I again 'met' with the two mentioned pioneers. I had then started at NTR, read the fundamentals of traffic theory and joined a project team analyzing the telephone network of Oslo, the so-called ARON-project. This project was a collaboration between STK and NT, represented by NTR, Technical Department and Region Oslo. One of the activity leaders was Eliot Jensen, then at STK, who for me became like a tutor in the teletraffic area. The strange thing was that the very first technical paper I was presented to by Eliot in this project was, in fact, an article by Palm: *Calcul exact de la perte dans les groupes de circuits échelonnés* [1]. Here, Palm discusses the solution of a simple grading with the help of state equations. No closed form solution was given by Palm, and I set out to solve the problem by generating functions. It turned out, though, that no general closed form could be found by this technique, either. Hence, I ended up solving the whole thing numerically on a computer. This possibility was definitely not easy at hand in the thirties when Palm looked upon the problem!

2 Conny Palm

Conny Palm was one of the most brilliant scientists in the history of teletraffic theory. Or perhaps I should say, *in the history of queuing theory,* since Palm really laid the mathematical fundamentals of this theory. He was born in 1907, joined the Royal Institute of Technology in Stockholm in 1925, finished, in practice, his studies in 1931 but did not take his final exam in electrical engineering until 1940. For nine years he was thus what we would call an eternal student!

However, his nine extra student years were by no means lazy years. From his publication list we find seven papers dated 1940 or earlier. From 1934 he worked within the area of telephony and joined Ericsson in 1936. He was an active participant at Cramér's seminars on mathematical statistics at Stockholm's University where he in 1937 met William Feller. This meeting turned out to be an important one for further development of the theory of stochastic processes; it was a meeting between an academic scholar searching for the fundamentals of probability theory and a 'practical' engineer who happened to have many of the answers. Palm had obtained these answers through his own work and the work of his predecessors within telephony. According to Cramér: "Palm presented a unique, mature and clarified view on his ideas of probability theory". And in fact, in 1950 Feller himself wrote: "Waiting time and trunking problems for telephone exchanges were studied long before the theory of stochastic processes was available and had a stimulating influence on the development of the theory. In particular Palm's impressive work over many years has proven useful to several authors" [2].

The culmination of Palm's work on the fundamentals of probability theory was his famous thesis in 1943: *Intensitätsschwankungen im Fernsprechverkehr.* This work has later been translated into English by Dr. Christian Jacobæus, previous technical director of L.M. Ericsson and one of the pioneers in the teletraffic community, with the title: *Intensity Variations in Telephone Traffic* [3].

The thesis consists of two theoretical parts that lay the fundamental basis for the last section in which he discusses various aspects of measurements. In the first part he makes a general study of point processes and their applications to teletraffic. Here he introduces the concept of *equilibrium points,* or in modern technology *regeneration* or *recurrence* points, that has later on proven extremely useful for the succeeding generations of traffic researchers. This general part has had a great influence on the theory of stochastic processes. The second part is a more specialized study of traffic with intensity variations, i.e. where the call intensity λ is a function of time, $\lambda(t)$. Here he discusses *slow* as well as *rapid* traffic variations.

Professor Arne Jensen at IMSOR in Copenhagen wrote an interesting article about Palm and his work which was presented at the 7th International Teletraffic Congress in Stockholm in 1973 [2]. Further details about Palm and his work can be found there. I would just like to add that Palm died in 1951 at the age of 44; that during the last four years of his life he was in charge of a project which resulted in the first Swedish digital computer, the so-called BARK; and that he was busy constructing a larger computer when illness prematurely ended his life. His last years were difficult ones on the personal level, a matter professor Jensen dwells upon in the above mentioned paper.

3 A simple grading

As mentioned in the introduction, the first work of Palm to which I was confronted at NTR was a treatment, in French, of a simple grading depicted in Figure 1. The grading consists of two separate groups (splits) with n_1 and n_2 circuits, respectively, and a common overflow group consisting of m circuits. Two independent traffic streams with call intensities λ_1, λ_2 are hunting for outlets in the separate splits as a *first choice.* If all of the outlets in either split are occupied, the arriving calls may hunt for a free outlet among the m of the common group.

When we talk of 'solving the grading', we really mean solving for the state probabilities $P(j_1,j_2,i)$, where j_1, j_2, i stands for j_1 occupations of split 1, j_2 occupations of split 2, and i occupations of the common group. The state equations given by Eqs (1) were numerically solved at NTR by using Gauss-Seidels' method and the successive relaxation method of Young [4].

The state equations of such a grading can easily be established. The main equations, i.e. the equations for the cases where none of the groups are fully occupied, take a simple form as seen in Eq (1a). Here A_1, A_2 is the offered traffic to the respective splits.

$$\begin{aligned}(A_1 + A_2 + j_1 + j_2 + i)P(j_1,j_2,i) \\ = (j_1 + 1)P(j_1 + 1,j_2,i) + \\ (j_2 + 1)P(j_1,j_2 + 1,i) + \\ (i + 1)Pj_1,j_2,i + 1) + A_1P(j_1 - 1,j_2,i) \\ + A_2P(j_1,j_2 - 1,i) \end{aligned} \tag{1a}$$

$$j_1 = 0,1,\ldots,n_1 - 1,$$
$$j_2 = 0,1,\ldots,n_2 - 1,$$
$$i = 0,1,\ldots,m - 1$$

What complicates the picture, and makes the general solution difficult to obtain, is the large number of boundary equations. There are six different types of boundary equations depending on whether one or more of j_1, j_2 and i take on their maximum values: $j_1 = n_1$, $j_2 = n_2$, $i = m$. For instance, for

$j_1 = n_1$, $j_2 \neq n_2$, $i \neq m$, we have:

$$\begin{aligned}(A_1 + A_2 + n_1 + j_2 + i)P(n_1,j_2,i) \\ = (j_2 + 1)P(n_1,j_2 + 1,i) + \\ (i + 1)P(n_1,j_2,i + 1) + \\ A_1P(n_1 - 1,j_2,i) + A_2P(n_1,j_2 - 1,i) \\ + A_1P(n_1,j_2,i - 1) \end{aligned} \tag{1b}$$

and similar for the other 5 equations. Needless to say, the set of equations can be rather complicated to solve for large n_1, n_2 and m.

By the way, I can mention that I was visited by Dr. L.A. Joys in those days concerning mathematical solutions of the very same grading. Joys, who at that time was a retired regional director of Norwegian Telecom, Bergen, had received his doctor degree in teletraffic theory a couple of years earlier with some very interesting work on the Engset and Erlang formula [5], and was now interested in finding solutions for the grading. He had started with small n_1, n_2 and m (like 1 and 2), solved the state equations for these simple cases and successively increased the number of outlets. However, he soon ran into mathematical problems and came to NTR for help. But when he understood that we had a *computer program* that could solve for any configuration of the grading, he quickly lost interest. Some times computers can take the fun out of things!

4 Research on telephone traffic carried by full availability groups

4.1 Some propositions regarding flat and steep distribution functions

My next 'meeting' with Palm was through some papers published in a special issue of Tele (Informations from the Royal Board of Swedish Telecommunications [6]) that were, in fact, English translations of some previously published articles in Swedish dating back to 1946. I found these articles extremely useful and used them extensively during my first years at NTR. One should bear in

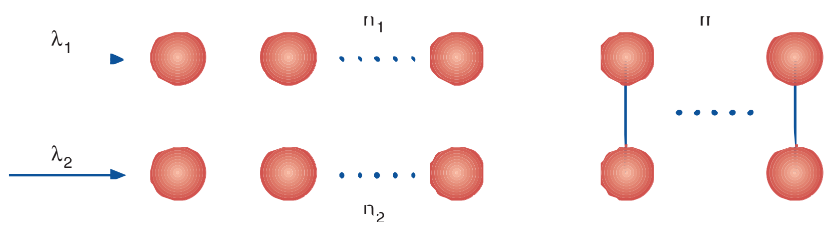


Figure 1 Simple Grading

mind that the seventies were before the PC-revolution, and we were in need of simple methods and algorithms in our daily work. And that was just what Palm provided us with through these papers.

The first article is named "Some propositions regarding Flat and Steep Distribution Functions" and deals with distributions usually arising in teletraffic; i.e. distribution functions limited one way, e.g. for time, $t \geq 0$. Here, he introduces the concept of *form factor* defined as the ratio between the second and the square of the first moment:

$$\varepsilon = \frac{\int_0^\infty t^2 f(t)dt}{\left\{\int_0^\infty tf(t)dt\right\}^2} = \frac{2}{s^2}\int_0^\infty tG(t)dt \quad (2)$$

In Eq (2) *f(t)* is the density function, *G(t)* the (decreasing cumulative) distribution function, and *s* the mean value of the stochastic function under consideration.

The form factor is closely related to the *peakedness,* η, defined as the square of the ratio between the standard deviation, σ, and the mean: $\eta = \sigma^2/s^2$. Hence, it is easily seen that

$$\varepsilon = 1 + \eta \quad (4)$$

The reason for Palm to introduce the form factor, rather than stick to the peakedness, is to work with parameters centered *around the zero* axis rather than around mean values. Whether this is an important point can be questioned; my experience is that the peakedness factor usually gives the same information.

For the exponential function the form factor equals 2. Other distributions are classed into two types, *steep* and *flat* according to whether the form factor is smaller or larger than 2:

Steep distributions:	$1 \leq \varepsilon < 2$
Flat distributions:	$\varepsilon > 2$

This classification has become important to all of us working in traffic theory and analyzing measured data. From the knowledge of the form factor we immediately know what type of distributions to look for in the process of data-fitting.

For instance, if we know that some measured traffic data has a form factor $\varepsilon > 2$, a natural choice for fitting would be a *sum* of exponential distributions. If *G(x)* is a distribution function with a density function *g(x) = -G'(x)*, we would for instance choose

$$g(x) = \sum_{i=1}^{n} \alpha_i \lambda_i e^{-\lambda_i x} \quad (5)$$

with $\Sigma\alpha_i = 1$ and $\alpha_i > 0$.

It can easily be proven that *g(x)* is a flat distribution. Physically, Eq (5) can be interpreted as a process with *n* types of events that all follow a negative exponential distribution with mean $1/\lambda_i$. Palm also proves that *G(x)* is a completely monotone function and that all such functions have a density function expressed as the Stieltjes integral:

$$g(x) = -G'(x) = \int_0^\infty ye^{-xy}dF(y) \quad (6)$$

where *F(y)* is a distribution function.

In the same way he demonstrates that the *convolution* of exponential distributions:

$$f_n(x) = \prod_i^n \otimes \lambda_i e^{-\lambda_i x} \quad (7)$$

is a steep distribution, i.e. with form factor < 2.

If all the λ's are equal, the convolution calculations in Eq (7) gives the famous Erlang distribution:

$$f_n(x) = \frac{\lambda(\lambda x)^{n-1}}{(n-1)!}e^{-\lambda x} \quad (8)$$

It has a mean n/λ and variance n/λ^2. Hence, we readily see that $\varepsilon = 1 + 1/n$, which is less than 2 for all *n* larger than 1, the latter being the simple exponential distribution.

Palm also discusses the interesting difference between the two classes of distributions discussed above concerning the *mean residual duration*. By this is meant the mean value of the remaining holding time that has lasted a time *t*. Or, in mathematical terms:

$$\mu_r = \frac{\int_0^\infty G(t+x)dx}{G(t)} \quad (9)$$

One might believe that this mean residual duration would decrease with *t*, which would mean that the longer one waits, the greater would be the chance of being served. This is far from always being the case, however, and Palm demonstrates that for *steep distributions* the mean residual duration indeed *diminishes with time* whereas for *flat distributions* the residual duration in fact *increases* with time!

4.2 Some observations on the Erlang formulæ for busy-signal system

The next paper in this collection deals with properties of the Erlang first formula $E_{1,n}$, i.e. the congestion probability in a full availability system with no queueing. Here, Palm derives many useful properties of $E_{1,n}$ and at the same time gives some interpretations of various traffic parameters. In fact, this article is in itself a very nice introduction to teletraffic theory.

The results are rather simple derivations, but they proved extremely useful for us working with traffic problems in the seventies. I remember I had the $E_{1,n}$ formula stored on magnetic cards for use on pocket calculators like HP 25. The calculations then were simply done by Palm's recursion formulas. For an indication of the type of formulas he gives, I quote the following:

First of all, the Erlang first formula is:

$$E_{1,n} = \frac{\frac{A^n}{n!}}{1 + A + \frac{A^2}{2!} + \ldots + \frac{A^n}{n!}} \quad (10)$$

From this formula we easily find the two following recursion formulas:

$$E_{1,n} = \frac{AE_{1,n-1}}{n + AE_{1,n-1}} \quad \& \qquad\qquad (11a)$$

$$E_{1,n-1} = \frac{n}{A}\frac{E_{1,n}}{A(1-E_{1,n})}$$

With $E_{1,0} = 1$ the result for arbitrary *n* can easily be calculated on a calculator.

Palm also gives the following formula for the derivative of $E_{1,n}$ with respect to *A*:

$$\frac{dE_{1,n}}{dA} = \frac{n - A + AE_{1,n}}{A}E_{1,n} \quad (11b)$$

From this formula can easily be seen how the relative increase of loss depends on the relative increase in traffic.

There are several other formulas in this paper, for instance some nice approximations. today, however, these results are probably most interesting from a historical point of view.

4.3 Contributions to the theory on delay systems

The last three articles in the series are devoted to delay systems. The first of them gives similar formulas for delay systems that was given for the pure loss system in the previous section:

The congestion formula, $E_{2,n}$, originally given by Erlang in 1917, can be expressed:

$$E_{2,n} = \frac{\frac{A^n}{(n-1)!}}{\sum_{i=0}^{n-1} (n-i)\frac{A^i}{i!}} \tag{12}$$

From this formula Palm deduces the following recursion relations:

$$\begin{aligned} E_{2,n} &= \frac{A(n-1-A)E_{2,n-1}}{(n-1)(n-A) - AE_{2,n}} \\ E_{2,n-1} &= \frac{(n-1)(n-A)E_{2,n}}{A(n-1-A+E_{2,n})} \end{aligned} \tag{13}$$

It is also possible to express $E_{2,n}$ in terms of $E_{1,n}$. and vice versa:

$$\begin{aligned} E_{2,n} &= \frac{AE_{1,n-1}}{n-A+AE_{1,n-1}} \\ E_{1,n} &= \frac{(n-A)E_{2,n}}{n-AE_{2,n}} \end{aligned} \tag{14}$$

Palm next goes through the derivation of the M/M/n and M/D/n FIFO queues. Even though the derived formulas go back to Erlang [7], Molina [8] and Crommelin [9], Palm's logical derivations and interpreting comments are well worth reading. He also compares the waiting time distributions for *constant* versus *exponentially* distributed holding times for quite a large number of cases. He concludes that "... it was found that the differences in general are so small that they must be considered as lacking significance in the work of dimensioning. The difference will be smaller the larger the groups are. Only with very small groups will the difference be of such a magnitude that one possibly should take them into account."

It is also interesting to note that Palm in the same paper even includes treatments of queueing system with *voluntary departure of those waiting,* i.e. queueing systems where the subscribers get impatient and replace the receiver. Whether the deducted formulas originally go back to Palm himself is a bit unclear, but the only reference he gives is to a paper by himself from 1937 [10]. It is, however, rather typical for Palm to write long papers where he includes several different topics. The practice of today would have been to chop them up into several independent publications!

The last two papers in the series are: "Waiting Times with random served Queue" and "Duration of Congestion States in Delay System". They are both long papers, perhaps a bit lengthy, with detailed derivations and several numerical calculations often presented graphically in several curves. Personally, I never used the results from the papers so I renounce from making any further comments.

5 Intensity variations in telephone traffic

5.1 Palm's dissertation and Dr. Jacobæus

As mentioned in the introduction, Palm wrote his famous doctor thesis with the original German title: *Intensitätsschwankungen im Fernsprechverkehr* in 1943. This fundamental work has since been considered as the bible of teletraffic theory. I will at this point refrain from going through this work in detail and rather limit myself to a few personal comments connected to it. For an extensive discussion I refer to the previously mentioned work of Arne Jensen [2].

To me, the thesis of Palm will always be closely associated with one of the great pioneers of teletraffic theory, namely Dr. Christian Jacobæus. I met Jacobæus for the first time in 1976 at ITC 8 in Melbourne. He was then still Technical Director at L.M. Ericsson and was well known to me through his famous works within *link systems* that dated back to the early fifties [11]. In fact, the so-called 'chicken diagrams', that is an instructive way of depicting the hunting path through a link system, go back to Jacobæus and is still in use. Incidentally, Palm himself acted as a tutor for Jacobæus' own thesis, just a year prior to his (Palm's) death.

Hence, it was with great respect I met with the famous Jacobæus at that time, myself being a newcomer in teletraffic research. But Jacobæus was always interested in new people and new ideas coming into his field of interest, and did, in fact, spend some time with us newcomers. I still remember the nice evening at the Sydney opera that culminated in a dinner for the "Nordic group" hosted by Jacobæus on behalf of L.M. Ericsson.

Since then, I happened to meet Jacobæus several times over the years. He retired from Ericsson just after the Melbourne ITC but showed up regularly on the ITCs and the Nordic Teletraffic Seminars (NTS). He showed genuine interest in any event that occurred within the teletraffic field in Scandinavia. I remember particularly in 1982, when Bjarne Helvik at ELAB was about to defend his doctorate, that Jacobæus showed up unexpectedly the night before at the hotel in Trondheim where Gunnar Lind and myself were discussing the question to be asked the next day. He firstly participated actively in going through the dissertation and secondly, acted as kind of a self-appointed third opponent during the very defense itself! I was told that something similar happened the year before, in 1981, when Arne Myskja defended his doctorate.

The reason for mentioning Jacobæus in connection with the thesis of Palm is because he was the one to translate it into English. The translation was finished in 1987, the book was published in 1988 and thus made Palm's work known to the scientific community the world over. I remember, though, that when I met Jacobæus in Stockholm around 1981 he told me already then that he was going to translate Palm's dissertation into English, and whenever I met him in the following years he talked about the progress with his book. Hence, he must have spent about 6 years on the translation, or at least had it at the back of his mind during these years. But the thesis [3], in its translated version, contains more than 200 pages in double columns, and Palm's way of philosophizing definitely makes it hard to translate. Besides, Jacobæus was ill the last years before his death in 1988.

5.2 Intensity variations

As dwelt upon in the introduction, Palm's dissertation covers a broad range of subjects, and may in itself be considered as a textbook in teletraffic theory. His style is not very modern, though, and he likes to give lengthy arguments and explanations before starting his mathematical treatments. His terminology might also differ somewhat from the one of today. Hence, one has to read the text carefully to fully grasp the underlying idea. This certainly must have been a challenge for the translator.

In telephony it is usually assumed that the call arrivals are distributed *randomly* over the time axis, and when one speaks of random traffic it is tacitly assumed that the arrival of a new call is independent of previous calls. Palm then argues that this assumption does not correspond with the real, observed traffic to a sufficient degree. To overcome this discrepancy, he introduces the concept of *intensity variations*. By this is meant that traffic intensities like *call arrival intensity* λ, the *releasing rate* μ, etc., are functions of time. The exponential distribution may thus be written $exp[-\lambda(t)x]$ where $\lambda(t)$ is the intensity at time *t*. We thus no longer have a random traffic.

He further distinguishes between *slow* and *fast* variations. By *slow* variations he roughly means that the traffic condition at every point of time in a statistical sense is the same as for a completely random traffic. That is, at a certain point in time we just use the intensity, *y(t),* at that point and otherwise work as before. In cases when the impact of the transient phenomena at a change of the intensity cannot be neglected, the traffic process will be described as *traffic with fast variations.* We will in the following consider slow variations only.

First of all Palm establishes general expressions for *time mean* functions and *call mean* functions. *Time mean* expresses the fact that a traffic function is observed (as from an outside observer) over a long period of time and the average value is calculated based on the relative time spent in each possible state. Call mean is the average value as seen from an arriving call.

Let *G(x)* be the distribution function for the intensity *y*, i.e. $Pr\{y < x\} = G(x)$. Let further *F(x)* be a general, time mean, function that is valid for random traffic with call intensity *x*. Then

$$\int_0^\infty F(x)dG(x) \tag{15a}$$

expresses the corresponding mean value for traffic with slow variations that for random traffic is expressed by *F(x).*

In the same way, if *f(x)* is a *call mean* function, the corresponding mean for traffic with slow variations is:

$$\frac{1}{y}\int_0^\infty xf(x)dG(x) \tag{15b}$$

By means of Eqs (15), all results obtained for random traffic can be applied for slow variations.

To take an example: The congestion in a group of *n* devices is $E_{1,n}(A)$ with *A* the product of mean holding time *s* and call arrival intensity λ. The corresponding time congestion for an arrival intensity with slow variation will be:

$$\int_0^\infty E_{1,n}(sx)dG(x) \tag{16}$$

For random traffic $E_{1,n}(A)$ also expresses the call congestion, i.e. the relative number of blocked calls. From Eq (15b) we, however, find the *call congestion* for traffic with slow variations:

$$\frac{1}{\lambda}\int_0^\infty xE_{1,n}(sx)dG(x) \tag{17}$$

Hence, for traffic with slow variations, the call congestion is different from time congestion.

Let us now go back to the questions discussed in section 4.1. We know that a *flat* distribution may be approximated by a sum of exponentials as given by Eq (5). However, a more general form is given by the Stieltjes integral in Eq (6). If we now assume that the intensity *x* is a variable, not necessary by time, distributed according to *F(y)*, we can find an alternative way of expressing a flat distribution. In fact, in many cases this new form is far easier to work with than the sum of exponentials.

As an example, let us assume that $f(y) = -F'(y)$ is a Poisson distribution[1]

$$f(y) = \tau_0\frac{(\tau_0 y)^n}{\Gamma(n+1)}e^{-\tau_0 y}, \quad n \equiv \frac{\tau_0}{\tau} \tag{18}$$

Here τ is the mean value (holding time, interarrival time etc.), τ_0 a fitting parameter and $\Gamma(z)$ the gamma function defined by

$$\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}dx \tag{19a}$$

with the property

$$\Gamma(z+1) = z\Gamma(z) \tag{19b}$$

Inserting *f(x)* into Eq (6) and using Eqs (19) gives

[1] *Palm uses the name Poisson distribution. As a time distribution, we would today rather call it an Erlang n+1 distribution.*

$$g(x) = \left(\frac{1}{\tau}+\frac{1}{\tau_0}\right)\frac{1}{\left(1+\frac{x}{\tau_0}\right)^{\frac{\tau_0}{\tau}+2}} \tag{20}$$

The distribution Eq (20) has a form factor

$$\varepsilon = \frac{2}{1-\frac{\tau}{\tau_0}} \tag{21}$$

which is always greater than 2 for $\tau_0 > \tau$. This last restriction is not really severe; since we are considering flat distributions in the first place τ_0 has to be larger than τ. Palm argue that the distribution Eq (20) is much simpler to work with for fitting than the sum of exponentials. This is, of course true, but the question is whether this really matters today with PCs and available software programs.

6 References

1 Palm, C. Calcul exact de la perte dans le groupes de circuit échelonné. *Ericsson Technics,* 3, 41–71, 1936.

2 Jensen, A. *Conny Palm : proceedings ITC 7,* 121/1–121/6, Stockholm, 1973.

3 Palm, C. Thesis : intensity variations in telephone traffic. *North-Holland Studies in Telecommunication,* 10, 1988.

4 Young, D. Iterative methods for solving partial differential equations of elliptic type. *Trans. Am. Math. Soc.,* 76, 92–111, 1954.

5 Joys, L A. *The Engset and Erlang formulae for telephone traffic losses.* Kjeller, Norwegian Telecom Research, 1971. (TF report 25/71.)

6 Palm, C. Special issue of Teletrafikteknik. *Tekniska Meddelanden från Kungliga Telegrafstyrelsen,* 1946.

7 Erlang, A K. Solution of some probability problems of significance for automatic telephone exchanges. *Elektroteknikeren,* Copenhagen, 1917 (Danish).

8 Molina, E C. Application of the theory of probability to the telephone trunking problems. *Bell system technical Journal,* 6, 1927.

9 Crommelin, C D. Delay probability formulae when the holding times are constant. *Post office electr. eng. j.,* 25, 1932–33.

10 Palm, C. Some investigations into waiting times in telephone plants. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* Stockholm, 7–9, 1937 (Swedish).

11 Jacobæus, C. *A study on congestion in link systems.* Thesis : Royal Technical University of Stockholm, 1950.

7 Appendix

Publications by Palm

In this section we give a complete list of papers published by Palm. The list is taken from [2].

1 Calcul exact de la perte dans le groupes de circuit échelonné. *Ericsson technics,* 3, 41–71, 1936.

2 Inhomogeneous telephone traffic in full-availability groups. *Ericsson technics,* 1, 36, 1937.

3 Några undersökningar över ventetider vid telefonanlegningar. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 7–9, 109–127, 1937.

4 Étude des délais d'attente. *Ericsson technics,* 2, 39–58, 1938.

5 Analysis of the Erlang traffic formulae for busy-signal arrangements. *Ericsson technics,* 4, 39–58, 1938.

6 *Några undersökningar över trafikförhållandena vid stel koppling.* Unpublished, 1938. (Telestyrelsen Sua 717.)

7 Telefonanläggningarnas betjäningskvalitet. *Ericsson review,* 3, 66–76, 1940.

8 Mätnoggrannhet vid bestämning av trafikmängd enligt genomsökningsförfarandet. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 7–9, 97–115, 1941.

9 *Speciella trafikmätningar vid Östermalms telefonstation 1941. Mätningarnas syfte och allmänna uppläggning.* Unpublished, 1941.

10 Intensitätsschwankungen im Fernsprechverkehr. Thesis : *Ericsson technics,* 44, 1943.

11 En formfaktor för bedömning av väntetidsfördelingar. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 1–3, 1–6, 1943.

12 Några följdsatser ur de Erlangska formlerna. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 1–3, 6–8, 1943.

13 Samtalsminut eller trafikenhet. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 7–9, 133–137, 1943.

14 Nyare rön inom trafikforskningen. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 4–5, 55–99, 1944.

15 Nya metoder för undersökning av telefontrafikens egenskaper. *Teknisk tidskrift,* 18, 557–561, 1944.

16 *Intensitetsvariationernas inverkan vid trafikmätningar.* Unpublished, 1945. (Telegrafstyrelsen Sua 695.)

17 – 21 Special issue of Teletrafikteknik. *Tekniska meddelanden från Kungliga Telegrafstyrelsen.* 109, 1946.

17 Some propositions regarding flat and steep distribution functions, 3–17.

18 Some observations on the Erlang formulae for busy-signal systems, 18–36.

19 Contributions to the theory on delay systems, 37–67.

20 Waiting times with random served queue, 68–87.

21 Duration of congestion states in delay systems, 88–106.

22 Spridning hos trafikmätvärden. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 3, 127–135, 1946. (English transl.: Fluctuations of measured telephone traffic. *Tele,* English Edition, 2, 1957.)

23 Teletrafikforskningen i Sverige. *Ingeniøren,* 37, E113–E115, 1946 (Danish).

Conny Palm

24 Registrering av samtal. *Tekniska meddelanden från Kungliga Telegrafstyrelsen,* 1, 1–15, 1947.

25 The assignment of workers in servicing automatic machines. *Journal of industrial engineering,* 9(1), 28–42, 1958. (Translated from Swedish.)

26 *Table of the Erlang Formula.* L M Ericsson, Stockholm, First ed. 1947, Second ed. 1954, Third ed. 1964.

27 Väntetider vid variabel anropsintensitet. *Ericsson review,* 4, 1947. (English transl.: Waiting times when traffic has variable mean intensity. *Ericsson review,* 4, 102–108, 1947.)

28 Metoder för bedömning av spörrningsolägenheter. *Tele,* 4, 189–208, 1952. (English transl.: Methods of judging the annoyance caused by congestion. *TELE,* English ed. 2, 1–20, 1953.)

29 *Elektrisk anordning för automatisk registrering, beräkning och indikering av resultat vid statistiska prov.* Svenska patentet nr 131,283. 1947.

Architectures for the modelling of QoS functionality*

BY FINN ARVE AAGESEN

A *service* is the behaviour of some functional capability. *Quality of service* (QoS) is a measure of the relative frequency of specific events within a service. These specific events are used as quality criteria for proper service functioning.

The *functional architecture* of a telecommunication service providing system is defined as the total set of functional elements and the dynamic relationship between these functional elements. This architecture has both an *operational* and a *management* part. The operational architecture defines the primary functionality related to the real-time handling of a call, while the management architecture represents the additional functionality needed for the administration of the operational functionality. A *QoS architecture* is a view of the functional architecture, focusing on traffic-resource-related aspects. A *QoS service* is defined as the QoS-related aspects of a service relationship. An optimum traffic solution is related to the existence of an optimum QoS architecture.

This paper discusses state-of-the art for QoS architectures for telecommunication service providing systems. The ISO/OSI and the ITU-F/ISDN QoS frameworks are presented. Some reflections on the ideal QoS architecture are also given.

Keywords: Computer-Communication Networks, Network Architecture and Design, Network Protocols, Telecommunication Networks, B-ISDN, Quality of Service.

1 Introduction

1.1 QoS in perspective

During the last few years, an increased attention has been given to the concept of Quality of Service (QoS). For telephone networks and circuit-switched data communication networks, QoS has always been taken seriously. During several decades, there has been a significant traffic research related to the QoS aspects of these systems, and the system design, standardisation and planning cultures have been QoS-oriented. The distances between the traffic research, system design, standardisation and traffic planning cultures have been relatively short.

As the packet switching technology was introduced in the 1970s, this co-operative QoS-oriented culture was not inherited. Considering the standardisation, QoS concepts have been defined and discussed, but many QoS-related items are either "for further study", "left to the network service provider", or not mentioned at all. The driving force behind packet-switching has been *service- and protocol-oriented* rather than QoS-oriented. There has been a significant traffic research related to packet switching. However, there are gaps between the traffic research, the system design and standardisation and the traffic planning cultures.

The B-ISDN specification and standardisation process has shown similar symptoms. It is well-known that ATM is much more vulnerable to bad dimensioning and non-optimal functionality than packet-switching. There has been a considerable traffic research related to the traffic aspects of ATM. However, there has been a gap between the traffic research and the system specification and standardisation cultures.

So, how come that QoS has not been appropriately handled for packet-switching and ATM? One obvious explanation is that QoS is difficult to handle. The appropriate handling requires a total view of the user behaviour parameters as well as an understanding of the various aspects of the system functionality. One other explanation is that the objective of the protocol-related standards is to focus on the interaction aspects between systems and on the information that is exchanged. QoS-related aspects can therefore fall out of the standardisation context. Protocol data units (PDUs) related to traffic-resource allocation are of interest, but *locally* applied functionality for the allocation of traffic resources are not. In a way QoS has *not been important enough*. It seems now, however, that the time is mature for the handling of QoS in a more systematic way and from the view of totality.

One reason for this increased focusing on QoS is that the explosion in the use of Internet has demonstrated the insufficiency of QoS-less protocols in high load situations. Internet is periodically so overloaded that the network is practically useless for planned professional work. A second reason is the awareness of the QoS-related problems faced in the ATM standardisation. ATM is by politicians and industry allotted the role of the backbone of the future Information Super Highway. QoS concepts are given much attention in the ATM standards. However, there is a lack of conclusions. If the QoS problems of ATM are not solved, ATM can adopt the nature of the connection-less Internet or the circuit-switched X.21, with upgraded bandwidths. A third reason for the increased focus on QoS is the growing number of applications with well-defined QoS requirements. The obvious inadequacy of the OSI transport layer to handle the well-defined QoS responsibility has stimulated the QoS discussion. The new evolving set of applications are represented by the "multimedia culture". This culture, seeing the communication system from above, has also contributed with their own solutions to many of the non-solved problems of packet-switching and B-ISDN.

So it seems that time wants QoS *solutions* rather than "for-further-study" *classifications.* The QoS-oriented culture that has been typical for the circuit-switched communication era may get its renaissance in the high capacity and multimedia communication era.

1.2 Outline of this paper

A summary of state-of-the art for QoS architectures for telecommunication service providing systems is given. It will be focused on the QoS aspects of the *primary operational* functionality of a system in a proper functioning state.

Section 2 gives a presentation of generic concepts. Concepts such as *service, QoS, QoS service, QoS architecture* and *traffic handling functionality* are defined.

Section 3 presents the ISO/OSI view of QoS, which is defined within the framework of the hierarchical OSI model. The OSI QoS Framework that is now under work is a valuable contribution to the introduction of QoS in packet-switched and ATM systems. However, so far, the framework is preliminary, and the intention must be followed up in the specific standards. Section 4 presents the narrowband and broadband ISDN views of QoS, defined within the framework of a user-network relationship.

In Section 5, the characteristics of an ideal QoS architecture is discussed. An application-oriented *QoS service* is an important element within an ideal QoS architecture. Section 6 gives summary and conclusions.

* *Part of the work reported in this paper was done under contract with Telenor Research as a part of the project "QoS in Layered Architecture".*

2 Generic concepts

2.1 Quality of service (QoS)

A *service* is the behaviour of some functional capability provided by a service provider to a service user (Figure 2.1). The service is defined by *service primitives*, and these service primitives carry *service parameters*. Within the context of ISO/OSI, a service is related to a specific layer, and an (N)-service is the service offered to the (N+1)-layer. The (N)-service is the result of the functionality of the (N)-layer-entity and the (N-1)-service. An (N)-service is offered at an (N)-service access point. ITU defines *teleservices* and *bearer services*. A teleservice is a service that the user gets from the user terminal, while a bearer service is a service offered at some interface between the user and the network. The ITU services represent OSI (N)-services at specific (N)-service access points.

ISO/OSI defines Quality of service (QoS) as *"a set of qualities related to the provision of an (N)-service, as perceived by an (N)-service-user"*. QoS is defined by ITU as *"the collective effect of service performance which determine the degree of satisfaction of the user of the service"*. The relative frequency of specific events within a service is used as a QoS measure. These specific events are used as quality criteria for the proper service functioning.

The ISO definitions are found within the OSI QoS Framework [27]. The ITU definitions are found in E.800 "Quality of Service and Dependability Vocabulary" [17] and in I.350 from the series of ISDN recommendations [22]. In addition to the service concept, ITU also has the concept of network performance (NP) defined as *"the ability of a network or network portion to provide the functions related to communication between users"*.

The concept *traffic performance* is often used in relationship with or instead of QoS. QoS is directly related to the use of common traffic resources. Examples of *traffic resources* are: nodes, transmission capacity, transmission links, routes, logical channels, buffers, windows, and also processing resources such as CPUs, buses and interface-circuits within nodes and end-systems. So the quantitative measure of QoS is directly related to the utilisation of the resources involved in providing the service, i.e. the traffic on these resources. So, *traffic performance* and *QoS* are two strongly related concepts.

A *QoS service* is here defined as the QoS-related aspects of a service relationship. While a service comprises the total functionality of some capability, the QoS-service only comprises the aspects of a service that have meaning for the definition of the QoS. The QoS-service defines the nature of the QoS parameters carried on the service primitives. The QoS-service is relevant between adjacent layers, but is also relevant for general client-server relationships.

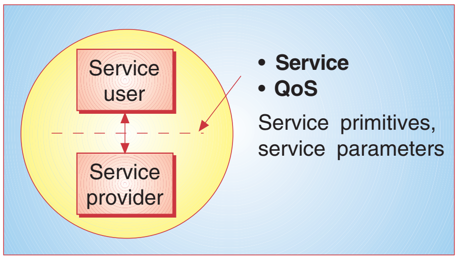


Figure 2.1 A generic service relationship

2.2 QoS architecture

The *functional architecture* of a telecommunication service providing system is defined as the total set of functional elements and the dynamic relationship between these functional elements. This architecture has an *operational* and a *management* part. The operational architecture defines the primary functionality related to the real-time handling of a call, while the management architecture defines the additional functionality needed for the administration of this operational functionality.

A *QoS architecture* is a view of a functional architecture, considering the traffic-resource-related aspects, i.e. the traffic resources and the functionality for the administration of these resources. Fault-handling functionality will be a part of the QoS architecture. This because of the tight connection between fault-handling and traffic resources. A QoS architecture will also have an operational and a management part, denoted as the *operational* and *management* QoS architecture, respectively. The relationship between the functional architecture and the QoS architecture is illustrated in Figure 2.2.

QoS is a *characteristic* of how well the service is performed. But a system must also have the ability to provide a certain QoS. This ability can depend very much on the system architecture and accordingly on the principles for the realisation of system functionality. The QoS archi-

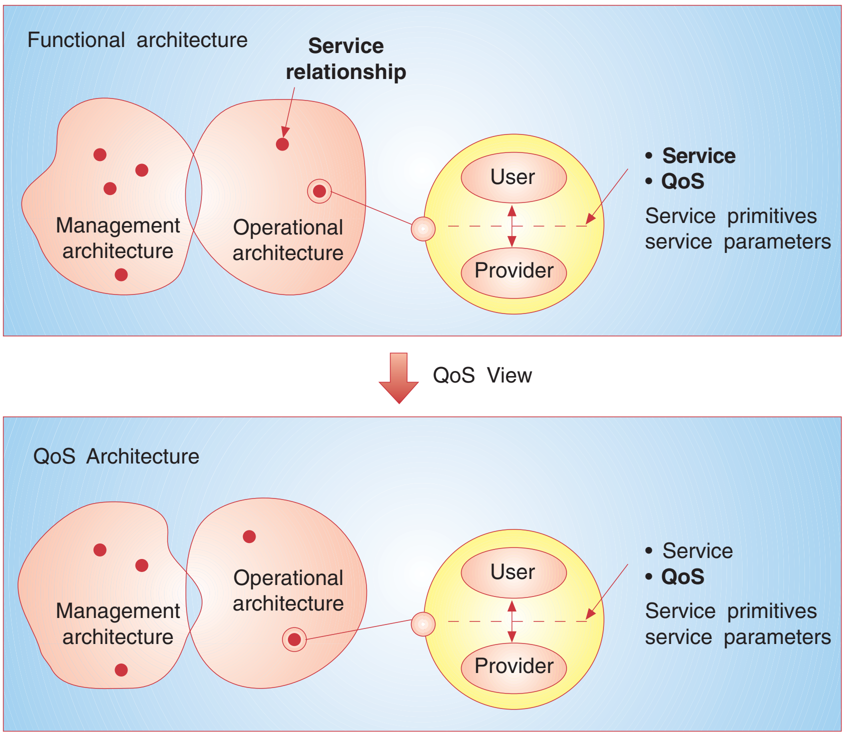


Figure 2.2 The relationship between the functional architecture and the QoS architecture

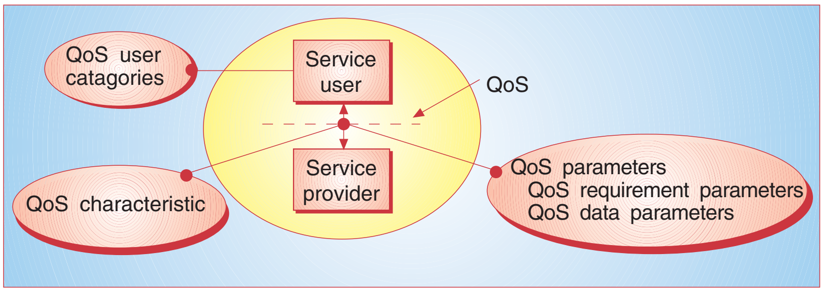


Figure 3.1 QoS user categories, characteristics and parameters

tecture focuses on how the QoS functionality is modelled. In the QoS architecture, QoS is the objective and the QoS-service is the solution. The service in the functional architecture is replaced by the QoS-service in the QoS architecture. The service parameters are replaced by the QoS parameters. An optimum traffic solution is related to the existence of an optimum QoS architecture, i.e. an architecture that gives the necessary possibilities without being too costly. In the B-ISDN standardisation, simplicity has been a strong requirement. Focus on simplicity, however, can lead to non-optimal solutions.

2.3 Traffic-handling functionality

The functionality for the allocation, administration and reallocation of traffic-carrying resources within the operational QoS architecture, is denoted as *traffic-handling functionality*. The *design objective* for the telecommunication services providing system is the offering of the defined set of services with a defined QoS at a minimum cost. The *QoS architecture* and the *traffic handling functionality* are tools for reaching this objective. The B-ISDN classification [8] of traffic-handling functionality is as follows:

- *traffic control:* the set of actions taken by the network to avoid congested conditions
- *congestion control:* the set of actions taken by the network to minimise the intensity, spread and duration of congestion.

In networks "up to LAN", the traffic control functionality was: *switching, multiplexing, routing, access, priority control* and *ack-based flow-control*. Traffic handling functionality for high capacity networks has been a topic for discussion for several years. The traditional *ack-based* stop-and-wait and sliding window flow control mechanisms are in the general case insufficient. With high-capacity networks came transport protocols with *rate-based flow-control*. In this case the number of information units over a predefined time period is controlled, rather than the number of outstanding acknowledgements.

With B-ISDN came concepts such as: *connection admission control* (CAC), *usage/network parameter control* (UPC/NPC) and *traffic shaping*. CAC is the set of actions taking place during call set up in order to establish a *traffic contract* and a connection, UPC/NPC is the set of actions taken by the network to monitor and control traffic, and traffic shaping comprises modification of the traffic characteristics. *Network resource management, priority control* and *feedback control* are also defined as part of the B-ISDN traffic handling functionality. Network resource management is the allocation of resources in order to separate flows according to service characteristics and feedback controls are the set of actions taken by the network and by the users to regulate the traffic submitted on ATM connections according to the state of network elements.

3 The ISO/OSI QoS framework

The Basic Reference Model for Open Systems Interconnection (OSI) defined in ITU recommendations X.200 and ISO/IEC 7498-1 provides a description of a model and the activities necessary for systems to interwork using communication media. The ISO/IEC "Quality of Service Framework" [27] will be a supplement to the description of QoS contained in this Basic Reference Model. In addition to the definition of QoS-related *concepts,* it contains the definition of a *model* of QoS and the definition of QoS parameter *semantics.*

The present version of the Quality of Service Framework still has holes. The document is formally and compactly written. The following presentation in the Sections 3.1–3.4 will therefore focus on the basic contents of this document rather than details. Section 3.1 presents QoS *user categories, characteristics and parameters,* Section 3.2 QoS *parameter semantics.* Section 3.3 presents *the model of QoS for OSI.* Section 3.4 is not directly related to the Quality of Service Framework document, but discusses QoS in existing layer-specific OSI recommendations.

3.1 QoS user categories, characteristics and parameters

The concepts QoS user categories, QoS characteristics and QoS parameters are essential in the OSI QoS framework. The relationships between these concepts are illustrated in Figure 3.1.

A *QoS user category* is defined as "a policy objective that leads to the identification of a set of QoS characteristics". The basic idea is to identify various classes of users and to define the QoS requirements based on these classes. A *QoS characteristic* is defined as "a quantifiable aspect of QoS, which is defined independently of the means by which it is represented or controlled". QoS characteristics are intended to be used to describe the actual behaviour of systems. A *QoS parameter* is defined "as a variable relating to one or more QoS characteristics, values of which are conveyed between objects as a part of a QoS mechanism". QoS parameters are classified as *QoS requirement parameters* and *QoS data parameters. QoS requirements* are a statement of requirements for one or more QoS characteristics.

The defined fundamental QoS user categories are: the *secure,* the *safe,* the *time critical,* the *highly reliable,* the *easy to use,* the *extensible/flexible,* the *low cost* and the *QoS monitorable/testable/auditable* user categories. Most of these categories are self-explanatory. The QoS monitorable/testable/auditable user category is related to the requirements for monitoring, testing and auditing of the QoS for the provided service. QoS *auditability* means that the service user must be able to obtain verifiable evidence of the QoS as actually provided and of the cost incurred as related to the service provided and to the quality of that service. The requirement for audit of the service provided may be met by the provision of suitable information on 1) the service as actually provided, 2) the QoS as actually provided, and 3) the cost incurred as related to the service and QoS. Concerning QoS characteristics, the following *generic* QoS characteristics are defined:

- *Time delay* represents the difference in time between two related events. The time delay characteristic is quantified in any units of time such as seconds, milliseconds, etc.
- *Time variability* represents the variability of a time or a time period. It relates to the dispersion or jitter that can be tolerated when time periods are involved.
- *Time window* represents specific period of time. It is a bounded time interval which is defined by a starting time and a time delay, by a starting and an end time or by a time delay and an end time.
- *Capacity* represents the amount of service that can be provided in a specified period of time.
- *Accuracy* represents the correctness of an event, a set of events or a condition. Accuracy is a QoS characteristic of concern to the user, for whom this characteristic refers to the user information only.
- *Protection* represents the security afforded to a resource or to information. Protection is quantified as a probability of failure of the protection.
- *Cost* represents a means of assigning value to an object. Cost is measured in terms of currency/unit.
- *Priority* represents the importance of an object or the urgency assigned to an event.
- *Availability* represents the proportion of time when satisfactory service is available.
- *Reliability* represents the probability that accuracy will remain above a defined requirement (i.e. that failures will not occur).
- *Coherence* is the QoS characteristic that represents the degree of correlation between two or more objects (events, actions or information).

3.2 The model of QoS

The purpose of the model of QoS for OSI is to define the basic concepts necessary for the description of QoS as it applies to open systems. Figure 3.2 illustrates the external flow of QoS requirements in a confirmed (N)-service-facility. The model of QoS recognises two types of entities performing functions related to the operation and management of QoS:

- layer QoS entities
- system QoS entities.

This is the same classification as used in the ISO recommendations for specification of OSI Management in the Recommendations ISO 9595-96, 10165:1-4 and 10040. Layer QoS entities are associated with the operation of a particular (N)-subsystem, and system QoS entities have a system-wide role.

The *layer QoS entities* comprise: the (N)-service user, the (N)-policy-control-function ((N)-PCF), the (N)-QoS-control function ((N)-QCF), the (N)-protocol-entity ((N)-PE) and the (N-1)-service-provider. Figure 3.3 illustrates the total flow related to outgoing (N)-QoS-requirements. The (N)-PCF receives the (N)-QoS requirements submitted by the (N)-service-user and applies specific policies defined for the (N)-subsystem. Examples are policies related to time critical communications. (N)-QCF will decide if the QoS requirements can be met by the operation of an existing (N)-PE. If so, such an (N)-PE is selected, otherwise the call is rejected. The QoS requirements can be modified before sending them to the (N)-PE. The (N)-PE can also reject the received QoS requirements. In order to perform its function the (N)-QCF may also need to access information provided by the system QoS entities.

The *system QoS entities* comprise: the system policy control function (SPCF), the system QoS control function (SQCF), the system management agent (SMA), the system management manager (SMM) and the resource manager (RM). These system entities are primarily concerned with management functions, and the set of system and layer QoS entities described constitutes a functional decomposition of an open system for the purpose of describing *QoS management*. In [27] some figures illustrating the relationship between the layer and system QoS entities are given. Figure 3.4 illustrates the relationship between the layer and the system entities.

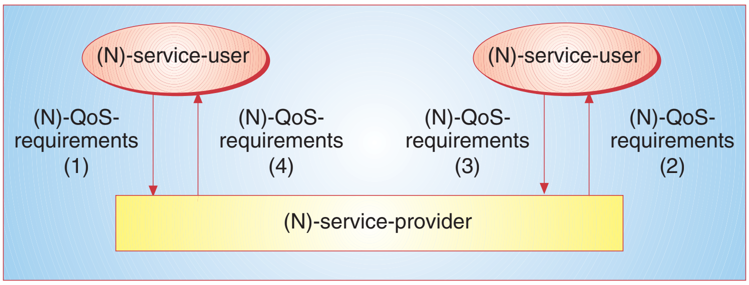


Figure 3.2 Flow of QoS requirements in a confirmed (N)-service facility (from [27])

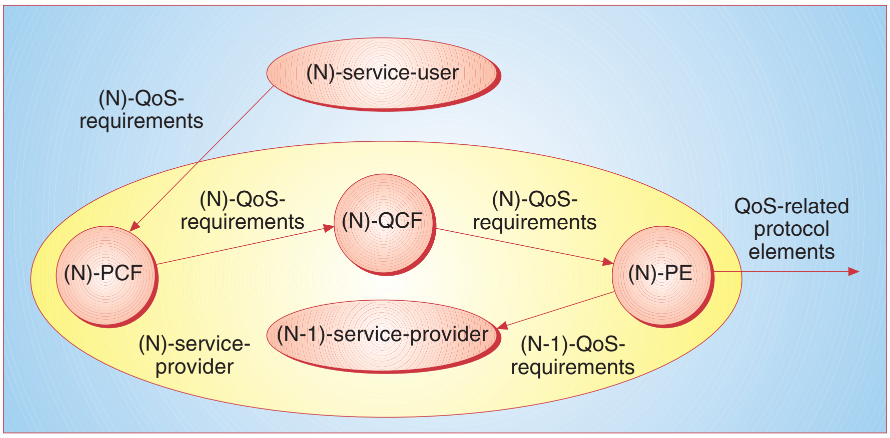


Figure 3.3 Outgoing flow of QoS requirements (from [27])

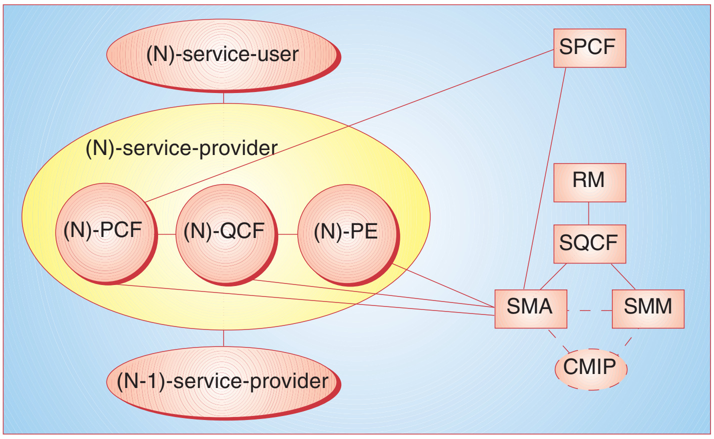


Figure 3.4 Relationship between layer and system QoS entities (from [27])

SMA represents the agent processing function of OSI systems management, which enables system resources to be remotely managed using OSI systems management protocols. It supports a set of system management functions, which will depend on the particular system configuration, and uses an OSI stack including CMIP (Common Management Information Protocol) for communication with remote SMMs. RM represents the end-system control of resources.

SQCF combines two functions: a system-wide capability for tuning the performance of the various protocol entities that are in operation, and providing co-ordination of any requirement to modify the behaviour of remote systems via OSI systems management. The SMM represents the operation of the open system in the manager role, as defined in the ISO/IEC 10040. Its inclusion in the model makes it possible to describe QoS management functions in which a system needs to control, or obtain information from, another system using OSI systems management protocols. Individual (N)-protocol-entities do not interact with SMM directly, but instead interact via the SQCF. This representation is chosen so that, where QoS is concerned, there is a single point of control of manager communication. Hence, the SQCF also has a co-ordination role.

The role of SPCF is basically similar to the role of (N)-PCF at each layer. The inclusion of SPCF recognises that any policy implemented at any particular layer is likely to depend on a policy which has been established for the whole open system. In particular in the case of a policy concerned with time critical communication, it is conceivable to consider an entity which has access to information concerning not just a single open system, but also other open systems involved in the time critical communication.

3.3 QoS parameter semantics

The QoS parameters have been classified as *requirement* and *data* parameters. However, there must be some semantics related to the defined parameter values. The part of the ISO/OSI QoS Framework handling parameter semantics is based on [14]. Some requirements are classified as *weak* (best effort) requirements and some as *strong* requirements. The following types are defined:

- *Target:* the value is a desired level of QoS
- *Lowest quality acceptable:* the value is a minimum level of QoS that is acceptable to the user
- *Guaranteed:* the desired QoS must be guaranteed by reservation of resources (or other means) so that the requested level will be met, barring "rare" events such as equipment failure
- *Compulsory:* the achieved QoS must be monitored by the provider, and the communication must be aborted if it degrades below the compulsory level. However, the desired QoS is not guaranteed, and indeed it may be deliberately degraded, for example to enable a higher precedence-demand for service to be satisfied
- *Threshold:* The achieved QoS must be monitored by the provider and either or both users warned if it degrades below the threshold level
- *Controlled highest quality:* The provider and users must constrain their operation so that the level is not exceeded – for example by delaying a PDU so as not to exceed a specified instantaneous throughput.

The "target" and "lowest quality acceptable" types are considered as the "best effort" semantics. In addition to these definitions, [14] gives example cases for various QoS parameter semantics for two-party and three-party cases. For one session, both "compulsory", "threshold" and "controlled highest quality" values can be defined for the same parameter. It is referred to [14] and Section 7.2 of [27].

3.4 QoS in present OSI recommendations

The concept of QoS is defined on various layers within the ISO/OSI reference model. In the following presentation the discussion is related to the QoS aspects of

- the application support part consisting of layers 5–7
- the transport layer, and
- the network layer and the link layer.

3.4.1 The application support part

The application layer consists of "Specific Application Service Elements" (SASEs) and "Common Application Service Elements" (CASEs). The SASEs are related to different application "cultures" such as the data-communication community, the telecommunication community, and also other application communities.

The data communication community is responsible for SASEs such as FTAM (File Transfer Access and Management), VT (Virtual Terminal), JTM (Job Transfer and Manipulation) and RPC (Remote Procedure Call). The telecommunication community is responsible for the newly defined INAP (Intelligent Network Application Protocol), while other application communities are responsible for SASEs such as EDI (Electronic Data Interchange), ELCOM (Electric Power Communication Protocol) and SGML (Standard Graphical Markup Language).

The CASEs are primarily defined within the data-communication community, and comprise elements such as ACSE (Association Control Service Element), ROSE (Remote Operation Service Element) and CCR (Concurrency, Commitment and Recovery).

No QoS parameters are defined on the services related to SASEs. The *only* application service element that has explicitly defined QoS parameters is the CASE element ACSE (Association Control Service Element). A QoS parameter field is defined on the A.ASSOCIATE service primitives. The A.ASSOCIATE primitive is mapped into the *presentation layer* P.CONNECT service primitive. There are no QoS parameters carried on the ACSE or the presentation layer PDUs (PDU = protocol data unit). The presentation layer maps the QoS parameters of the presentation layer P.CONNECT primitive into the QoS field of the session service S.CONNECT primitive. The session service QoS parameters are *passed further* to the transport layer. The QoS parameters of the S.CONNECT primitive are similar to the connection-oriented transport service QoS parameters defined in Table 3.1. So there is no QoS-related *protocol* functionality in the OSI layers 5–7. The QoS parameters in the ACSE service A.ASSOCIATE primitive are just sent further down to the transport layer.

3.4.2 The transport layer

The transport service provides transparent transfer of data between transport service users. It relieves these users from any concern about the detailed way in which supporting communication media are utilised to achieve this transfer. There is both a connection-oriented and a connection-less service, but only a connection-oriented protocol. Transport services are defined in [26] and the connection-oriented transport protocol in [25]. The transport service provides for a QoS selection: *"The transport layer is required to optimise the use of available communication resources to provide the QoS required by the communicating transport service users at a minimum cost. QoS is specified through the selection of values for QoS parameters representing characteristics such as throughput, transit delay, residual error rate and failure probability."*

The QoS parameters for the *connection-oriented* case is shown in Table 3.1. The concept "speed" used in Table 3.1 should be replaced by "time delay" according to the QoS Framework defined in the previous section. QoS is negotiated at connection set-up. There is not specified any particular values or classes of values for the QoS parameters. *Throughput* and *transit delay* are defined for each direction of transfer on a transport connection. In each direction both *maximum* and *average* values are specified. Both throughput and transit delay specifications are

Table 3.1 Connection-oriented transport service QoS parameters (from (26))

Phase	Performance criterion				
	Speed (Time delay)	Accuracy/Reliability	Others		
			Protection	Priority	Cost
TC establishment	TC establishment delay	TC establishment failure probability (misconnection/TC refusal)			
Data transfer	Throughput	Residual error rate (corruption, duplication/loss)			
	Transit delay	Resilience of the TC			
		Transfer failure probability			
TC release	TC release delay	TC release failure probability			

based on a stated average T-SDU size (T-SDU = transport service data unit).

The QoS parameters for the *connection-less* service are a subset of the connection-oriented parameters. They comprise *transit delay, protection, residual error probability* and *priority.* For details see Clause 10 and 17 of [26]. The QoS handling within the transport layer is important. Three issues are defined in the recommendation:

1 The QoS negotiation with the network layer

2 The QoS negotiation with the peer transport entity

3 Status exchanges and monitoring of QoS.

Concerning 1), the QoS parameters of the connection-oriented and connection-less network services are similar to the QoS parameters of the corresponding transport services, so that the transport service QoS requirements can be sent on to the network layer.

Concerning 2), the 5 classes of connection-oriented transport protocols [25] are classified according to type of underlying network and multiplexing and error recovery capability. For all classes except class 0, the connect request and connect confirm T-PDUs have the following QoS parameter fields: *maximum* and *average throughput* in both directions, specified as "target" and "minimum acceptable", *transit delay* and *residual error rate* in both directions, specified as "target" and "maximum acceptable" and finally *priority.* The classes 2–4 have an ack-based flow-control mechanism. The mapping of QoS requirement to window sizes should be a transport layer function.

Concerning 3), status exchanges and monitoring of QoS are not included in the present version of the transport protocol definition, but are covered by the QoS parameter semantics of the OSI QoS Framework briefly described in Section 3.3. In the present transport layer definition, the QoS handling is based on "best effort semantics".

3.4.3 The network layer and the link layer

As stated above, the network and transport service QoS parameters are similar. Considering the QoS functionality of the network layer protocols, very little is found except the ack-based flow-control mechanisms. For X.25, the "Essential Optional Packet Switched User Facilities" defined by X.2, defines the following optional facilities related to QoS: *throughput class negotiation, fast select* and *transit delay selection and identification.* Throughput class is a concept without exact semantic definition. It gives the user a possibility to select from a number of classes offering different throughput. Nothing is said about traffic handling functionality such as the mapping of QoS requirements to window sizes and call acceptance control. *ISO IP protocol* has an optional part in the header containing: *transit delay, protection from unauthorised access, cost determinants, residual error probability* and *priority,* corresponding to the QoS parameters of the connection-less network service.

For traditional *link layers* based on HDLC, there is not defined any link layer service, and HDLC has no PDU-fields related to QoS except the fields related to ack-based flow- and error-control. For link layers based on LLC/MAC, the LLC service primitives have a QoS *service*

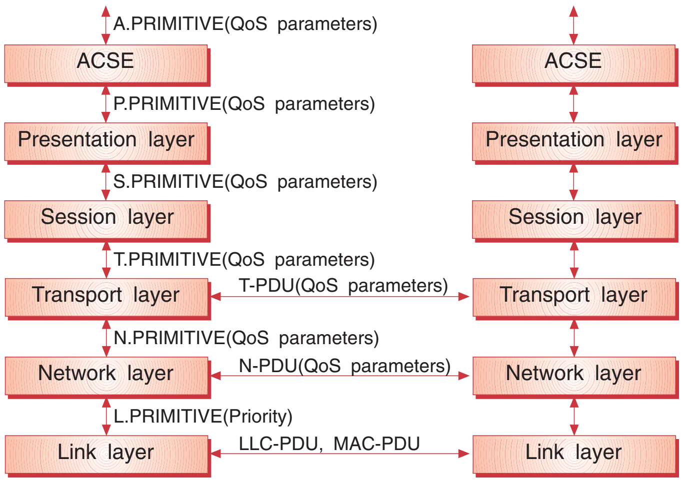


Figure 3.5 OSI QoS parameters in LAN-environments

class parameter field used to indicate the *priority.* There is *no* QoS parameter in the LLC PDUs except the flow- and error-control fields. There is also a *service class* field in the MAC service primitives used for priority. The use of this field is dependent on the available mechanisms in the specific protocols. The MAC-protocol provides the transmission capacity *access function*, and the various protocols have appropriate mechanisms and PDU-fields for this purpose.

3.5 The OSI QoS service

The aspects of QoS parameters related to service primitives and PDUs are illustrated in Figure 3.5. The OSI QoS-service can be characterised as *application-oriented.* This because the (N+1)-layer defines its QoS requirement parameters based on (N+1)-layer concepts, and there is an intention of a downwards mapping of the application-defined QoS requirements. There is a consistent set of session, transport and network service QoS requirement parameters.

There is a *concept of a totality,* but some important elements are missing: 1) The QoS parameters of the application layer service are not well-defined. 2) The transport layer is the vital point for handling the end-to-end QoS, but there is not defined any transport layer functionality to handle this responsibility except the delegation of this responsibility to the underlying network layer. 3) The network layer does not have the defined functionality to take this responsibility. 4) In addition, the need for parameters describing other aspects of the traffic pattern "offered" than the aspects reflected by the defined "requirements", is not explicitly expressed. These other aspects, however, can be defined as if they were QoS requirements.

4 The ISDN QoS framework

ISDN is here used as a common concept for narrowband (N-ISDN) and broadband ISDN (B-ISDN). The ISDN functional architecture consists of *the user plane* for data-transfer, the *control plane* for call and connection control and the *management plane* for management. Important QoS-related aspects are defined in the following documents:

- I.350: General Aspects of QoS and Network Performance in Digital Networks, Including ISDN
- I.356: B-ISDN ATM Layer Cell Transfer Performance
- I.321: B-ISDN Protocol Reference Model
- I.371: Traffic Control and Congestion Control in B-ISDN
- Q93B: User-Network and Network-Network Signalling
- ATM Forum: UNI 3.0.

This section will mostly comprise B-ISDN. Section 4.1 describes the generic QoS-aspects of ISDN and partly also the specific QoS-aspects of N-ISDN. In Section 4.2, the basic operational B-ISDN architecture is presented. Section 4.3 gives a short summary of ATM system service classes, protocol types and QoS classes. Section 4.4 defines traffic and congestion control functionality, and Section 4.5 defines traffic contracts. Section 4.4 and 4.5 are based on Section 3.6 of UNI 3.0 [8] and the ITU recommendation I.371.

4.1 The ISDN generic QoS framework

I.350 was intended for N-ISDN. The generic aspects of I.350 are also applicable to B-ISDN. The ITU concepts Quality of Service (QoS) and network performance (NP) were defined in Section 2.1. Figure 4.1 illustrates the concepts. *Performance parameter* is used as a common concept for QoS and NP parameters. A *primary* performance parameter is a parameter or a measure of a parameter determined on the basis of direct observations of events at service access points or connection element boundaries. A *derived* perfor-

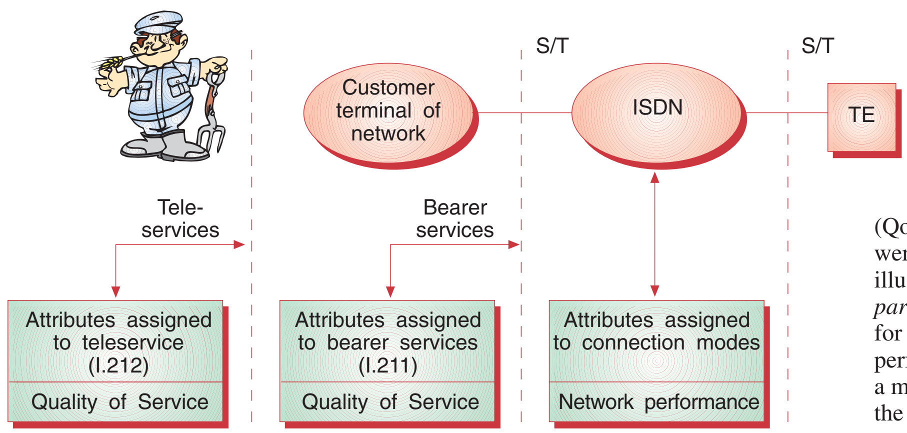


Figure 4.1 General scope of QoS and network performance (from I.350)

mance parameter is a parameter or a measure of a parameter determined on the basis of observed values of one or more relevant primary performance parameters and decision thresholds for each relevant primary performance parameter.

I.350 defines a *method* for identifying parameters. The “method” is rather a *matrix,* with the *functions:* access, user information transfer and disengagement as *rows* and the *macroscopic generic parameters:* speed, accuracy and dependability as *columns.* The following nine *generic primary* performance parameters are defined: *access speed, access accuracy, access dependability, information speed transfer, information transfer accuracy, information transfer dependability, disengagement speed, disengagement accuracy* and *disengagement dependability*. In addition, *availability* is defined as a *generic derived* performance parameter. In addition to these generic parameters, bearer service QoS parameters as well as circuit-switched and packet-switched NP parameters are defined.

Considering N-ISDN, the concepts of QoS and NP are not very much visible in the specific recommendations. Within the I.200-series defining *teleservices,* QoS is *for further study*. For the *circuit-switched bearer-service* capabilities, the recommendation I.352 specifies the objectives for the *connection set-up* delay and *disconnect delay*. For the *bearer-service* capabilities based on *packet-switching,* the recommendation I.122 “Framework for providing additional packet mode bearer services” adds the OSI network layer QoS aspects presented in Section 3.

The document I.356 “B-ISDN ATM Layer Cell Transfer Performance” is a supplement to I.350. The defined parameters apply to cell streams in which *all cells conform to a negotiated traffic contract* and are intended to characterise ATM connections in the *availability* state. The following performance parameters related to *cell transfer* are defined: *cell error ratio, severely-errored cell block ratio, cell loss ratio, cell misinsertion rate, cell transfer delay, mean cell transfer delay* and *cell delay variation.*

4.2 The B-ISDN functional operational architecture

A model for the B-ISDN operational functionality is shown in Figure 4.2. The basic elements are 1) the ATM system service user, 2) the ATM system user plane functionality, 3) the ATM system control plane functionality, and 4) the ATM network connection admission control function (CAC).

The User Network Interface (UNI) plays an important role within the B-ISDN architecture. The ATM system service user consists of two parts: the applications and the transport system. The *applications* comprise layer 5–7 functionality,

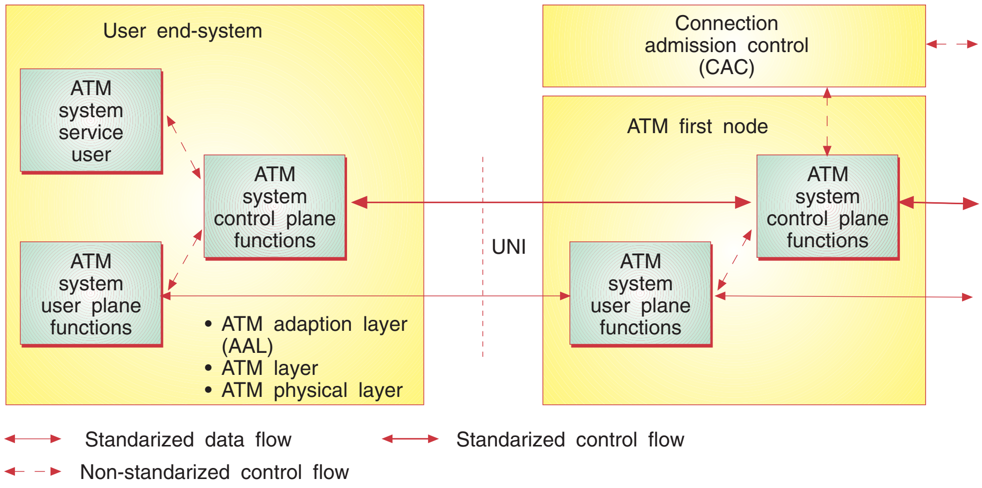


Figure 4.2 A B-ISDN generic operational architecture

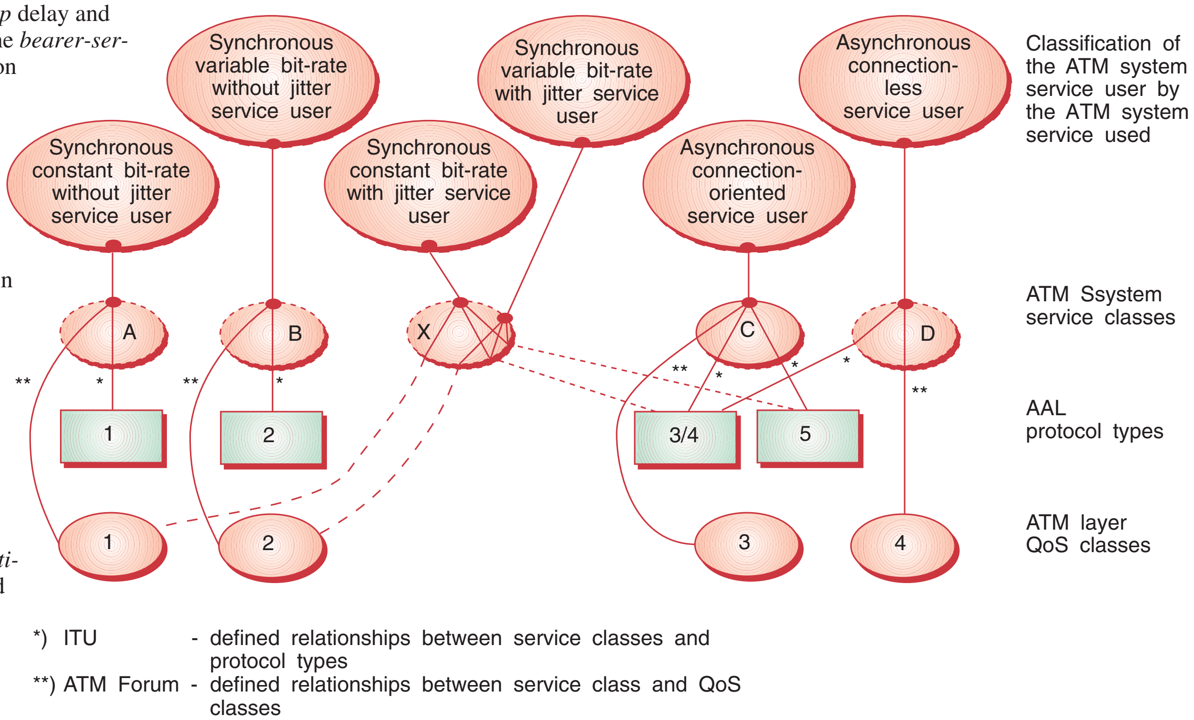


Figure 4.3 ATM system service classes, protocol types and QoS classes

while the *transport system* consists of layer 4 functions or combined layer 3–4 functions. The ATM system user plane functionality consists of the ATM adaption layer (AAL), ATM cell-transfer layer (ATM layer) and the ATM physical layer.

4.3 ATM system service classes, protocol types and QoS classes

The ATM system has five different service and protocol concepts. These are 1) the ATM system service classes, 2) the AAL protocol types, 3) the ATM layer protocol class, 4) the ATM system signalling protocol type and 5) the ATM layer specified and unspecified QoS classes. The relationship between the ATM system service classes, AAL protocol types and the specified QoS classes are illustrated in Figure 4.3.

There are four basic ATM system *service classes:* class A, B, C and D. Recently a new class X has been defined. The classes A, B and X are for *synchronous* applications, i.e. there are timing relations between sender and receiver. The service classes C and D support asynchronous applications, i.e. applications without timing relations. The synchronous classes A and B offer synchronisation capabilities. Class A provides a CBR service (constant bit-rate), class B a VBR service (variable bit-rate). Class C offers a general connection-oriented service and class D a connection-less service. The new class X is a specific service class for CBR and VBR synchronous applications that do not require synchronisation capabilities from the AAL layer. The ATM system *service class* is neither visible in any defined service primitive nor in any signalling information element. The service class is the resulting consequence of a selected ATM layer *QoS class* and an AAL *protocol type.*

The AAL protocol types are type 1, 2, 3/4 and 5. Type 1 supports a class A service, and type 2 supports a class B service. Types 3/4 and 5 support class C and also X services, while type 3/4 supports a class D service. The ATM system control plane uses a type 5 AAL protocol.

The *ATM QoS class* represents a QoS capability provided by the *ATM layer.* A QoS class can have *specified* performance parameters or *no specified* performance parameters. These cases are denoted as *specified QoS class* and *unspecified QoS class,* respectively. ATM Forum has defined four specified QoS classes, one for each of the service classes A, B, C and D. Future QoS classes supporting the same service class may be introduced. ATM QoS classes were introduced by ITU, but were left for further study.

4.4 ATM layer traffic and congestion control design objectives

The B-ISDN traffic and congestion control functionality was defined in Section 2.3. The stated objectives of ATM layer traffic and congestion control are: 1) to support a set of ATM layer QoS classes sufficient for all foreseeable B-ISDN services, 2) not to base the operation on AAL protocols which are B-ISDN service specific or on higher layer protocols which are application specific, but to let the protocol layers above the ATM layers use the information that is provided by the ATM layer to improve the utility those protocols can derive from the network, and 3) to design an optimum set of ATM layer traffic controls and congestion controls, minimising the network and end-system complexity while maximising the network utilisation.

4.5 The ATM traffic contract

A *traffic contract* specifies the negotiated characteristics of an ATM layer connection at a private or public UNI and is established between the user and the network before the establishment of a connection. The traffic contract comprises:

- Source traffic parameters
- Cell delay variation (CDV) tolerance
- Conformance definition
- Requested *QoS class* for each direction of the ATM connection
- The definition of a *compliant* connection.

The *peak cell rate* (PCR), the *sustainable cell rate* (SCR), and the *burst tolerance* (BT) are potential source traffic parameters. The contract items: PCR, SCR, BT, CDV tolerance and conformance are all based on i) a *generic cell rate algorithm* (GCRA) and ii) end-system *reference models*. GCRA consists of two parallel working algorithms: a virtual scheduling algorithm and a leaky bucket algorithm. GCRA has two parameters: *I* and *L*. *I* is the increment parameter (specified time between cells), and *L* the limit parameter (specified time period for accepting burst of cells). The virtual scheduling algorithm controls the time between individual cells, and the leaky bucket algorithm controls the accumulation of cells. The leaky bucket leak-rate is 1 per cell time, the limit is *L* and the increment per event is *I*.

PCR is defined according to the PCR reference model illustrated in Figure 4.4. PCR of an ATM connection is defined as the inverse of the minimum interarrival time *T* (peak-emission interval) between the appearance of two ATM-PDUs. The CDV tolerance is the upper bound on the "cell clumping" measure. The CDV tolerance allocated to a particular connection at the private UNI is denoted by the symbol τ^* and at the public UNI by τ. SCR and BT are defined according to the SCR and BT reference model similar to the PCR model illustrated in Figure 4.4. *Conformance* applies to cells as they pass

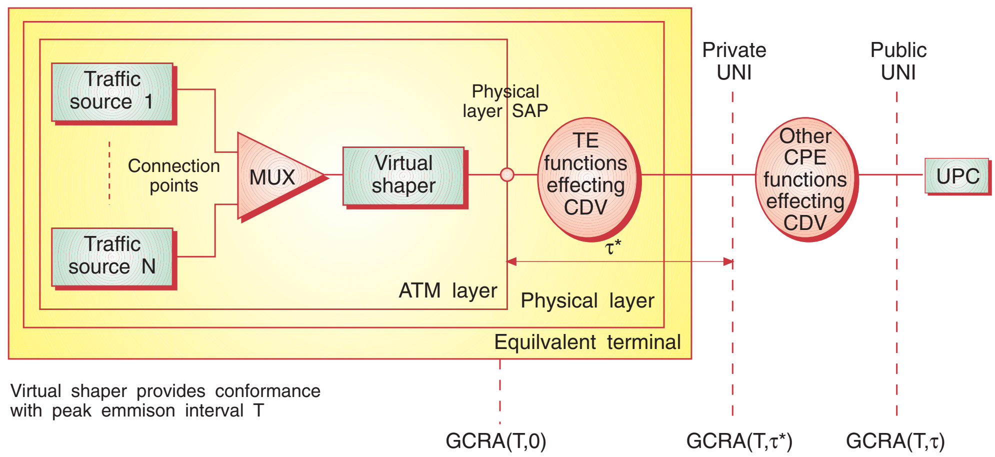


Figure 4.4 Peak cell rate reference model (from [8])

the UNI and are in principle tested according to some combination of GCRA algorithms. Conformance definition should *not* be interpreted as the Usage Parameter Control (UPC) algorithm. The network provider may use any UPC as long as the operation of the UPC does not violate the QoS objectives of compliant connections.

4.6 The B-ISDN QoS-service

The B-ISDN QoS-service can be characterised as *transmission-oriented*. The ATM system service class is the consequence of a selected ATM layer *QoS class* and an AAL *protocol class*. The ATM system user needs to know the concepts of cell-time, cell delay variation, burst tolerance and GCRA. The reference models are "just" reference models and *not* specification models. The functionality for providing GCRA(T,0) for PCR at the physical layer SAP is not defined in the standard. The virtual shaper is just a "thought example" [8].

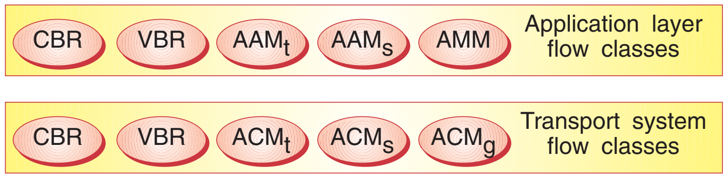
5 Some reflections on an ideal QoS architecture

The operational part of the OSI and ISDN QoS frameworks comprises:

- A *static architecture* defined by a set of entities performing *traffic handling functions*
- QoS-related parameters
- For B-ISDN, *a traffic contract* between the user and the network
- For OSI *an application-oriented* QoS-service based on QoS requirement parameters
- For B-ISDN a *transmission-oriented* QoS-service based on the *traffic contract*.

With this as a basis, some modifications as well as some new aspects are proposed. The following items are discussed:

- Flow-pattern-oriented view of the layered functionality (Section 5.1)
- Application-oriented QoS-service (Section 5.2)
- Time region concepts (section 5.3)
- Layered co-operation and layered traffic contracts (section 5.4).



CBR : constant bit-rate
VBR : Variable bit-rate
AAM : asynchronous association mode
AMM : asynchronous message mode
ACM : asynchronous connection mode

Figure 5.1 Application and transport layer flow classes

5.1 A flow-related classification of the layered functionality

The OSI layers and ISDN user plane are based on extensive multiplexing. The concepts ATM system *service class*, AAL *protocol class* and the ATM layer *QoS class* are flow pattern oriented. The concept *flow class* is hereby used as a generic concept for flow-pattern oriented service and protocol classifications. The basis for the resource allocation is the nature of the flow patterns. A QoS architecture therefore needs flow pattern concepts for the whole architecture. This subsection gives a flow-oriented view of the application layer and the transport layer, *application flow classes* and *transport system flow classes* are defined. The concepts used are illustrated in Figure 5.1. The *applications,* consisting of the functionality of the OSI layers 5–7, are classified as follows:

- Constant bit-rate flow class (CBR)
- Variable bit-rate flow class (VBR)
- Asynchronous association mode flow class (AAM) consisting of the
  - · transaction flow class (AAM$_t$), and
  - · the block-transfer stream flow class (AAM$_s$)
- Asynchronous message mode flow class (AMM).

"Asynchronous" means that the sender and receiver applications are not in real-time synchronisation. "Association mode" means that the sender and receiver applications have an "OSI application layer association". Class AMM represents *message-oriented* applications, i.e. one way information without immediate response. The transaction class AAM$_t$ contains both *interactive* and *distributed system* traffic. Distributed system traffic comprises traffic between application programs as well as operating system traffic. Definitions of QoS requirement and traffic flow parameters for the various flow classes are found in [3] and [4].

The *transport system* consists of layer 4 functions or combined layer 3–4 functions and will map the application classes to the ATM system service classes. Concerning various transport systems, we find:

- *traditional ISO transport system classes and TCP, both with ack-based flow-control and*
- *"new" transport systems with rate-based flow-control.*

The *traditional* transport systems are intended for both transaction traffic, block-transfer streams and message mode application traffic. Considering the *"new"* set of transport systems proposed for being used in high capacity networks, these do reflect the *nature* of various *application* classes. The following gives a non-exhaustive list: Delta-t, NETBLT (The Network Block Transfer Protocol), CMP (Check Mode Protocol), NADIR, VMTP (The Versatile Message Transfer Protocol), XTP (The Express Transfer Protocol), RTTP (The Real-Time Transport Protocol), TPR (Transport Protocol for Real-time Services) and TP++. TPR is an example protocol for the CBR application class, XTP for VBR, VMTP for AAM$_t$ and NETBLT for AAM$_s$. General overviews are given in [16], [15] and [19]. Details on VMTP can be found in [10], [23], NETBLT in [12], XTP in [24], [11], RTTP in [15], TPR in [7] and TP++ in [9]. Because of weak performance requirements, any transport system can be used for the application class AMM. The most likely choice, however, is a transaction-oriented or a block-transfer-stream-oriented protocol. Considering

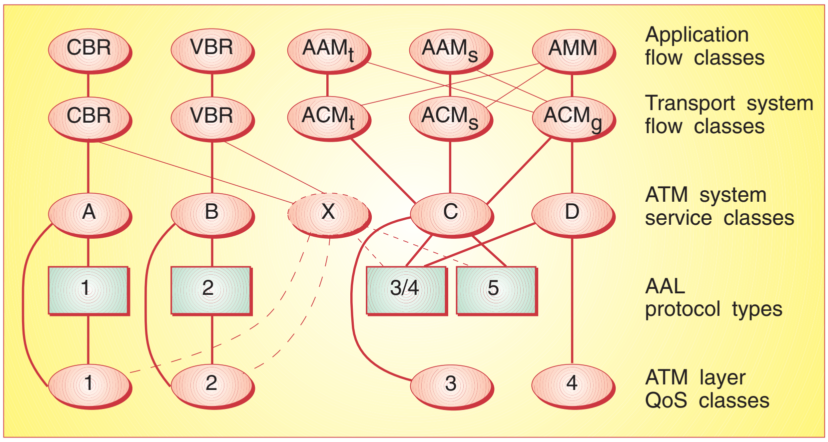


Figure 5.2 Layered flow class functionality

both the traditional ack-based and the "new" rate-based protocols, the *transport system flow classes* can be defined as:

- Constant bit-rate flow class (CBR)
- Variable bit-rate flow class (VBR)
- Transaction flow class (AAM_t)
- Block-transfer stream flow class (AAM_s)
- General connection flow class (AAM_c).

Figure 5.2 illustrates the application and transport system flow classes defined and also the relationship between these classes and the *specified* ATM system flow classes defined within the present B-ISDN QoS framework. The *unspecified* QoS class is not included in Figure 5.2.

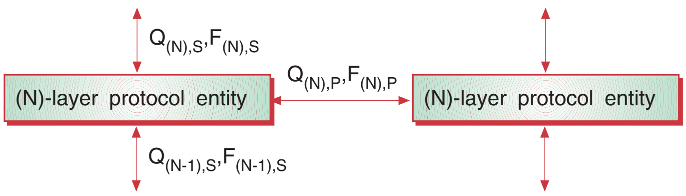


Figure 5.3 The F- and Q-parameters related to a layer

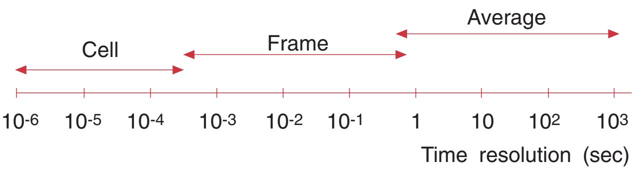


Figure 5.4 Time regions

5.2 Application-oriented QoS service

The OSI *QoS parameters* are QoS requirement and QoS data parameters. The ISDN *performance parameters* are QoS parameters, traffic parameters and NP parameters. These definitions are not consistent. A slightly changed definition is therefore used, where QoS parameters comprise QoS *requirement* and *traffic flow* parameters abbreviated to Q- and F-parameters respectively. We then have:

Q-parameters: For the definition of QoS requirements for individual information units

F-parameters: For the definition of the overall structure of the traffic flow pattern.

The index "S" indicates service, the index "P" indicates protocol. An additional index (N) indicates (N)-layer. See Figure 5.3. For a general (N)-layer, the following questions must be answered:

- Is there an externally defined (N)-service? What is the (N)-layer internal structure of flow classes, and what is the external visibility of these classes?
- Are the (N+1)-layer $F_{(N+1),P}$-parameters presented to the (N)-layer? If so, are the (N+1)-layer parameters related to an (N+1)-layer or an (N)-layer flow class?
- Similarly, are the $Q_{(N+1),P}$-parameters related to an (N+1)-layer or an (N)-layer flow class?

For the transmission-oriented ATM QoS-service the (N+1)-layer states the functionality it wants, based on knowledge about the (N)-layer internal flow class structure. The (N+1)-layer declares its F- and Q-parameters with reference to an (N)-layer flow pattern model. The alternative *application-oriented* QoS-service is to let layer (N+1) declare which flow class it represents and also its F-parameters based on the pattern of its own declared class. The external (N)-layer functionality can optionally be specified. For B-ISDN, a simpler ATM system flow class structure than the one illustrated in Figure 5.1 is then made possible.

Knowledge of the nature of the (N+1)-layer traffic pattern can be useful within the (N)-layer, both for *traffic handling* functionality and *traffic load estimations*. Using the (N+1)-layer flow pattern as the reference for the service will *conserve knowledge* about the application traffic pattern. This can also give the (N)-layer the possibility to utilise the freedom caused from lack of one-to-one mapping between the two flow classes. For ATM, traffic shaping based on cell spacing can be applied, utilising the freedom defined in the application and transport system F- and Q-parameters ([3], [4]).

5.3 Time regions

In a 160 Mbit/sec ATM network, the duration of a cell is 2–3 msec. The time resolution region related to the duration of single cells and sequences of cells is defined as the *cell time region*. Neither the *applications* nor the *transport system* will in the general case be able to operate in the cell time region. This fact is one of the reasons for defining various time regions to be used as a *dimension* for characterising the Q- and F-parameters and also the traffic handling functionality. In addition to the *cell* time region, the *frame* and the *average* time regions are introduced. The corresponding cell, frame and average *processes* describe the activities in the related time regions (Figure 5.4).

The average process is the flow pattern related to a pre-defined *average interval* in the order of seconds or higher. The concept *frame* is a generalisation of the concept video-frame, which is typically between 25 and 40 msec. A frame is a *significant application-related* quantity of information. The frame-time is the time for the "presentation" of this information. For image-related sources, the frame is one terminal screen. For sources with a "fixed" period, i.e. CBR, VBR and AAM_s, the frame-time is *the period* of the generated flow pattern. For the application classes with neither periodicity nor real-time demanding human users (ex: distributed system traffic and mail traffic), the frame has no direct intuitive interpretation.

The F-and Q-parameters for the (N)-service should be related to a time region applied in the (N+1)-layer operation. This is denoted as the QoS-service's *time region principle.* It is meaningless to specify (N+1)-layer F- and Q-parameters that does not match the time resolution used in the (N+1)-layer. A transport system working in the 100 msec time region should not be forced to specify its parameters within the cell time region.

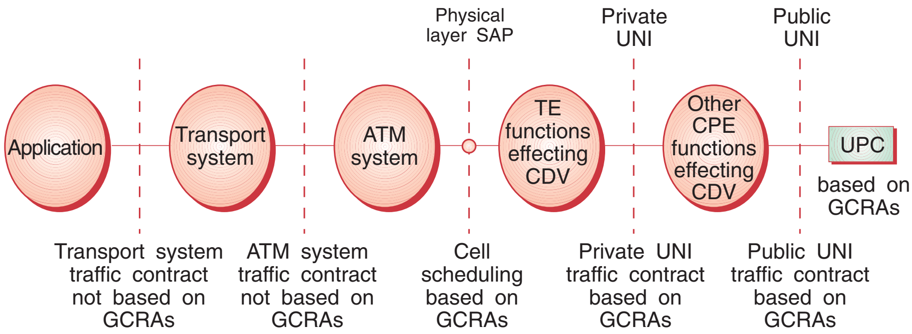


Figure 5.5 A layered traffic contract scheme for B-ISDN

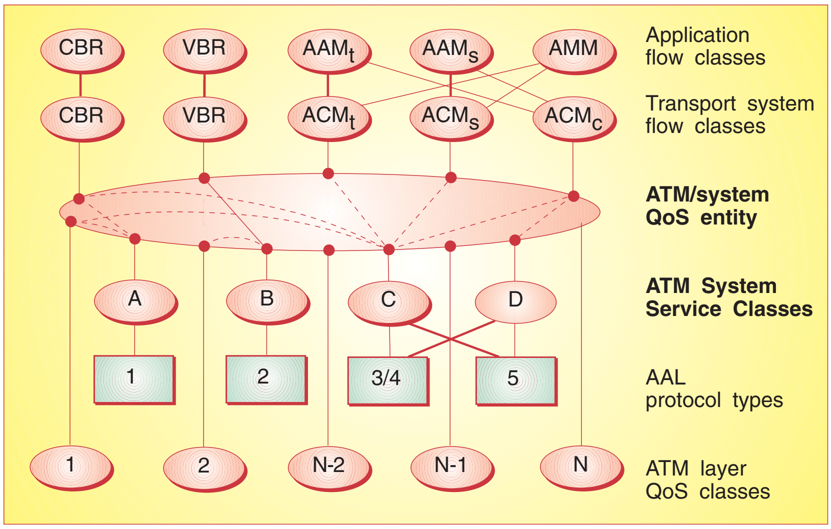


Figure 5.6 A revised B-ISDN flow class structure

5.4 Layered co-operation and layered traffic contracts

Layered co-operation means that the various elements of a communication task is delegated to various protocol layers. This is the basic idea behind layering. However, certain traffic handling functions such as *flow control* and *multiplexing* are *duplicated* rather than delegated. This is discussed by several authors ([28], [13] and [18]). By

- avoiding multiplexing above the link layer
- using an application-oriented QoS-service, and
- using the time region principle,

traffic handling functions such as flow control, usage parameter control and traffic shaping can be handled by *co-operation* ([3], [4]). As an example for B-ISDN, the transport system can do the traffic handling functionality in the average and frame time region. The ATM end-system must do the important mapping between the time regions and also the traffic shaping within the cell time region.

The B-ISDN concept *traffic contract* is hereby generalised. An *(N)-layer traffic contract* is defined as the *negotiated* set of F-and Q-parameters $\{Q_{(N),S}, F_{(N),S}\}$ of the (N)-layer service. The existing B-ISDN traffic contract is the realisation of such a *layer traffic contract* at the UNI.

If a traffic contract between the ATM system service user and the ATM system based on *frame* and *average* time region concepts is introduced, then the ATM system could define *another traffic contract* for UNI based on F-and Q-parameters in the *cell* time region. This contract is directly deduced from the higher layer contract.

UPC at the UNI with minimum uncertainty can only be obtained if the ATM system in the unit interfacing the UNI has responsibility for details of the contract. In this case the presented traffic load on the ATM links through the UNI can also be taken into consideration. Such a scheme is illustrated in Figure 5.5. With a layered contract scheme, there must be a well-defined ATM system service also comprising the control relationship illustrated in Figure 5.1. There must also be a well-defined traffic handling functionality in AAL.

A revised flow class structure based on an application-oriented QoS service is illustrated in Figure 5.6. By giving the responsibility for the selection of the internal ATM system flow class functionality to an ATM system QoS entity, a simpler application view of the ATM system is obtained. The internal ATM system flow class structure is also considerably simplified.

6 Summary

The operational part of the OSI and ISDN QoS frameworks comprises:

- A *static structure* of entities performing *traffic handling functions*
- QoS-related parameters, and
- QoS-services.

With this as a basis, some modifications as well as new aspects have been proposed. Within this paper, the QoS parameters have been classified as QoS requirement parameters (Q-parameters) and traffic flow parameters (F-parameters). The proposed solution is based on an application-oriented QoS-service formally defined by layered traffic contracts, consisting of a negotiated set of Q- and F-parameters. Time-regions are made visible as a dimension in the characterisation of Q-parameters, F-parameters and traffic handling functionality.

A telecommunication service providing system shall satisfy the requirements of the applications. A QoS architecture based on an application-oriented flow classification of the total layered functionality seems therefore to be the appropriate choice. This will conserve knowledge about the application Q- and F-parameters. By avoiding multiplexing above the link layer, QoS-related co-operation between the layers is made possible. Such a solution gives a flexible and flow-harmonised architecture.

References

1 Aagesen, F A, Helvik, B E. On excess process policing in B-ISDN networks. I: *13th International teletraffic congress (ITC 13).* North-Holland, Elsevier Science Publishers, 1991.

2 Aagesen, F A. *On source spacing in B-ISDN.* Trondheim, SINTEF, 1992. (Report STF 40 A92140.)

3 Aagesen, F A. A flow management architecture for B-ISDN. *IBCN&N,* Copenhagen. North Holland, Elsevier, 1993.

4 Aagesen, F A. *ATM system service primitives and flow management functionality : a design proposal.* Trondheim, SINTEF, 1993. (Report STF40 A93157.)

5 Aagesen, F A. *QoS in layered architecture.* Trondheim, SINTEF, 1994. (Report STF40 F94088.)

6 Aagesen, F A. On a quality of service operational architecture. *IFIP SEACOMM'94,* Kuala Lumpur, October 1994.

7 Anastasi, G et al. TPR : a transport protocol for real-time services in an FDDI environment. I: *International workshop on protocols for high speed networks*, *IFIP WG 6.1/6.4,* 1990.

8 The ATM Forum. *ATM user-network interface specification version 3.0,* September 10, 1993.

9 Biersack, E W et al. Gigabit networking research at Bellcore. *IEEE network,* 6(2), 1992.

10 Cheriton, D R. VMTP as the transport layer for high-performance distributed systems. *IEEE communication magazine,* June 1989.

11 Chesson, G. XTP/PE design considerations. *International workshop on protocols for high speed networks, IFIP WG 6.1/6.4.* Zurich, May 1989.

12 Clarc, D D, Lambert, M L, Zhang, L. *NETBLT : a bulk data transfer protocol.* Network Working Group RFC 998.

13 Clarc, D D. Architectural issues for a new generation of protocols. I: *Proceedings of Sigcomm 90,* September 1990.

14 Danthine, A et al. The OSI 95 connection-mode transport service : the enhanced QoS. I: *4th IFIP conference on high performance networking,* Liege, December 1992.

15 Dabbous, W S. RTTP : real-time transport protocol. I: *International workshop on protocols for high speed networks, IFIP WG 6.1/6.4.* Zurich, May 89.

16 Doeringer, W A et al. A survey of light-weight transport protocols for high-speed networks. *IEEE transactions on communication,* 38(11), 1990.

17 ITU-T. *E.800: terms and definitions related to the quality of telecommunication services, quality of service and dependability vocabulary.* Geneva, 1993. (ITU-T Blue Book, fascicle II.3.)

18 Feldmeier, D C. Multiplexing issues in communication system design. I: *Proceedings of the Sigcomm 90,* September 1990.

19 Hehman, D B et al. High-speed transport systems for multi-media applications. I: *International workshop on protocols for high speed networks, IFIP WG 6.1/6.4.* Zurich, May 1989.

20 CCITT. *Recommendations, general structure.* Geneva, 1988. (CCITT Blue Book, vol III – fascicle III.7, Part I – I.100-series.)

21 CCITT. *Recommendations, service capabilities.* Geneva, 1988. (CCITT Blue Book, vol III – fascicle III.7, part II – I.200-series.)

22 CCITT. *Recommendations, overall network aspects and functions.* Geneva, 1988. (Blue Book, vol III – fascicle III.8, part III - I.300-series.)

23 Nordmark, E, Cheriton, D R. Experiences from VMTP : how to achieve low response time. I: *IFIP WG 6.1/6.4, International workshop on protocols for high speed networks,* Zürich, May 1989.

24 *XTP protocol definition, rev. 3.5.* Protocol Engine Inc., September 1990.

25 ISO. *Revised text for ISO/IEC DIS 8073 – information technology – telecommunication and information exchange between systems – connection oriented transport protocol specification,* Geneva, 1991. (ISO/IEC JTC1/SC6 N6776.)

26 ISO. *Information technology – transport service definition for open systems interconnection.* Geneva, 1993. (ISO/IEC DIS 8072.)

27 ISO. *Quality of service framework, Working Draft no. 2.* SC21/WG1 meeting, Yokohama, 1993. (ISO/IEC JTC1/SC21.)

28 Tennenhouse, D L. Layered multiplexing considered harmful? I: *International workshop on protocols for high speed networks, IFIP WG 6.1/6.4.* Zurich, May 1989.

Observed traffic variations and their influence in choice of intensity measurement routine

BY ASKO PARVIALA

Abstract

In this paper the traditional understanding of stabilities and predictability of the traffic profiles and models have been weighted against the observations. In Helsinki Telephone Company, several wide scale traffic measurement studies were made in the 1980s, in order to expose how regular the traffic variations are. These measurements indicated that the conventional understanding about the stability of the repeating profiles, together with the popular traffic models during the busy hour, is dubious. This has, even in the international standardisation, caused a need to rethink the traffic intensity measurement routines, as a basis for the dimensioning of networks.

1 Measurements of traffic characteristics and dimensioning

The quality of telecommunications services depends on the network bottlenecks, whereas the overdimensioning of other parts in the network does not improve the service. Existing or expected bottlenecks are revealed by measuring the real traffic. Directly measurable are the *quality characteristics* observable by the subscriber; like congestion, waiting time, or failure rate. This information is used to control the offered service quality, but it is also necessary for immediate measures, such as fault hunting and repair. The quality characteristics are non-linear: they deteriorate fast as the load increases, approaching the network's capacity limits. So, these are not usable in forecasting. But the load, measured in erlangs, bits per second, etc., is linearly related to the use of the service, and thus forecastable, e.g. by time series. Therefore, the network is dimensioned for a *dimensioning hour* intensity, calculated according to certain rules from the measured peak intensities. In determining the required number of circuits or other capacities, a *mathematical model* is used, which combines the *intensity,* the nominal *quality norm* and assumptions of *statistical* properties of traffic (distribution of call intervals and holding times), concerning the network *structure* in question (single circuit-groups, alternative routing, waiting opportunities). It has been popular to assume a statistical equilibrium and the Poisson-distribution for the offered traffic during the dimensioning hour. Many academical theories have been created and presented on this or a slightly modified basis. How to constitute the dimensioning hour has met with less interest. The total service quality over a year is a result of the representativity of the dimensioning hour among all hours of the year, together with accuracy of the model, and the nominal congestion during that hour. It could be presented in number of minutes in the year without service, caused by the network's bottlenecks.

2 Measurements of intensity

The *measurement costs* concern the reserved equipment, the work of preparing and starting the measurements, and the logical handling, pruning and reporting of the measurement data. A balance will be searched between the costs of the carried measurements and the disadvantages caused by the intensities which are not measured. Among the disadvantages are unknown or unexpected breakdowns of the service quality, and unoptimal augmentations in the network.

There are two different approaches with the following measurement routines:

- *Scheduled routines* which minimise the measurement periods and the data to be handled by concentrating on the expected high value periods only. The nominal congestion is kept low in order to compensate the presumably lost peak intensities left outside the measurement periods. This also results in a high dependency on the mathematical model, and a high investment level due to the loose dimensioning. These routines are the traditional way of dimensioning, trying to minimise the use of separate measuring equipment, which is expensive in use due to lots of manual handling of data.
- *Continuous routines* which activate the measurements all the time. The abundance of data is pruned in the earliest phase by searching the relevant peak intensities. The nominal congestion can be higher while no compensation for lost peak intensities outside the measurement periods is needed. This results in smaller dependency on the mathematical model, and in a lower investment level due to the possibility of focusing augmentation of the network equipment correctly. These routines are in use today, the measurements using just the basic data produced all the time by the exchange control processor(s).

The worst choice is to create complex but inaccurate measurements, needing a low nominal congestion. Even this choice has been made by some operators.

Every *circuit-group* between two points has its traffic capacity, which depends on the number of its circuits. If it is a potential bottleneck, it must be measured distinctly. But if, instead of one, several circuit-groups in parallel, or an overflow cluster, or a network between two subscriber groups, is considered, then the traffic capacity of the studied circuit-group is no more constant, but lower, depending on the traffic caused by other circuit-groups. This complicates the dimensioning and the traffic management, calling for special measures by measurements. Complications of this kind are less common in a radial network structure, there including the cases where circuit-groups for reliability reasons are divided into two similar groups with symmetrically offered traffic.

In *comprehensive telecommunications courses* is commonly taught that the offered teletraffic consists of rather regular variations, obeying annual, weekly and daily rhythms, and random Poissonian distribution during the dimensioning hour, the *busy hour*.

If these assumptions are valid, it is sufficient to measure the busy hour intensities during a few days annually only – the rest of the year can be ignored – and to dimension by the Erlang formulae and their derivatives. The *elementary conceptions* in question concern the existence of the year's high season, its annual consistency, the peak-hour's consistency between seasons, representativity of the average day, the absence of peak loads outside the day's peak-hour, the statistical balance and Poissonian distribution during the peak-hour.

Both the rule to select the dimensioning hour and the mathematical model originated *early in this century*, from manual exchanges, and communication needs of the past society. They have sometimes been called into question, but they have been persistently favoured by some influential organisations. The new telecommunication networks, the new services and the nearly total renewal of users, have only to a limited extent influenced the common views.

While the word "busy hour" has been used ambiguously, it is avoided here by talking about the *peak-hour* as a post-selected hour having the highest inten-

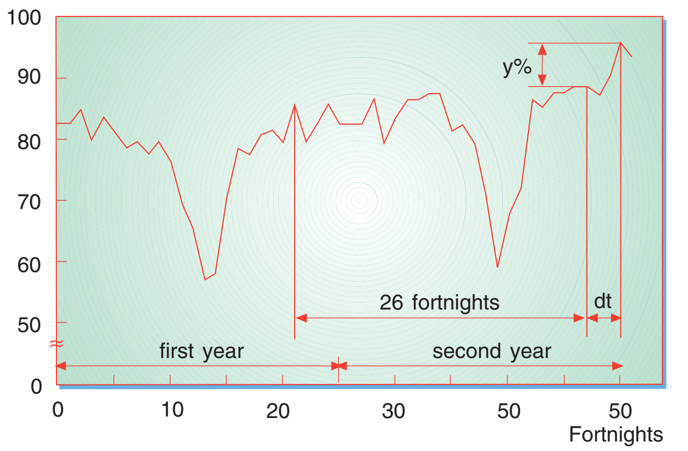


Figure 1 Illustration of two years of peak-hours' fortnight average intensities Y in one circuit-group, with its timing error dt and intensity error y %

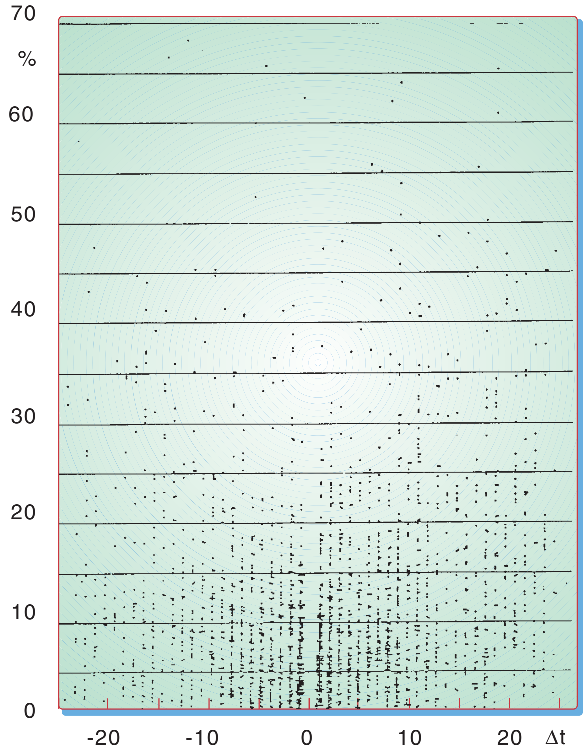


Figure 2 Timing errors dt and intensity errors y % of 2728 circuit-groups [5]. The 305 error-free cases are not indicated

sity, the *measurement period* as a pre-selected time for measurements, and the *dimensioning hour* intensity, created by certain rules and prognoses from the measured or estimated intensities.

In the choice of intensity measurement routine, the knowledge of traffic *profile stability* is important: if they are stable, the short pre-selected measurement periods save data handling capacities; if they are unstable, continuous measurement with data post-selection is needed.

In Helsinki Telephone Company (HTC), from the end of the 1970s to the year 1993, some intensity measurements have been carried out in order to *resolve the validity* of the common assumptions about stable intensity peaks, and ideality of traffic during the peak-hour. Since the measuring resources were convenient, it was possible to perform the measurements on a mass scale, not only taking a few samples. Thus, the results are widely representative, at least in the Helsinki circumstances. The measurements used the methods and routines which were in daily use in Helsinki; just the data processing afterwards was modified in order to simulate different measurement routines under study.

Several presentations in international fora are summarised in the following, and their most relevant results are referred to.

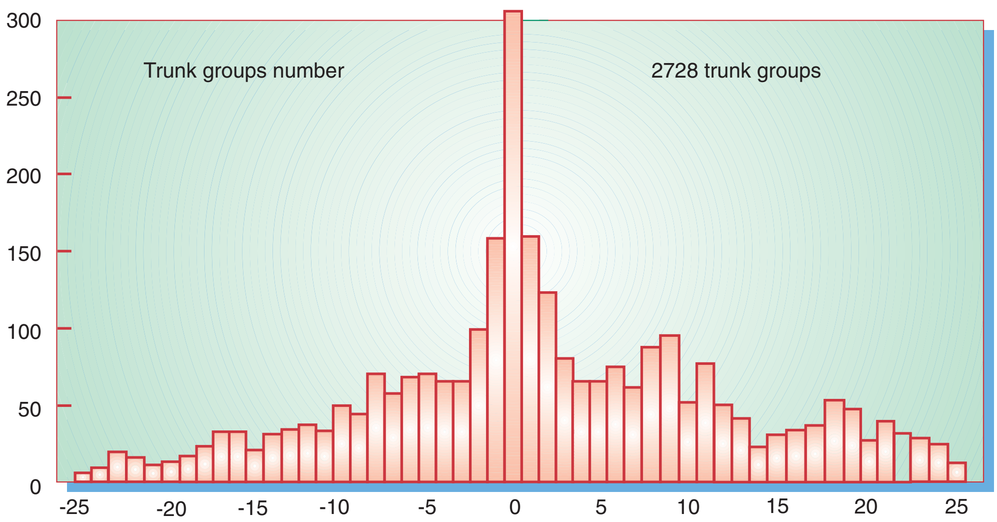


Figure 3 Distribution of timing errors dt [5]

As references are used the *Elementary Conceptions,* being formulated to support the traditional understanding and thus the high stability of profiles.

3 The high season and its stability

The *Elementary Conception (ElCo) about seasons* is that there is *in the year a season with higher traffic than other seasons, and it is stable from one year to another.* If the ElCo about seasons is valid, the intensity measurements can be concentrated to these seasons only, when a high load has been discovered in earlier measurements.

The *study object* in the Helsinki local network was the 2728 circuit-groups connected to the measuring system AUTRAX. It measured each circuit-group continuously, all year round, in ten consecutive working day rounds, i.e. in fortnight periods, each being registered as an average of the days' peak-hours. The measurement during two years, 1980 to 1981, thus included 52 fortnightly intensities. On an average a circuit-group had 62 circuits and a fortnightly intensity of 29.1 E. The measured traffic volume exceeded 250 MEh. The complete study is published in [5].

Concerning one circuit-group, the *first year's peak* load time was identified, and the expected time and intensity 26 fortnights later were searched. The expected values were compared with the real second year's peak load time and intensity. The timing error *dt* and the intensity loss *y* % characterise that circuit-group, Figure 1.

If the ElCo about seasons were valid, all the circuit-groups were concentrated to point $dt = 0$, $y\ \% = 0$. In fact, the ElCo about seasons was fulfilled only in a minority of cases, the rest indicating a wide scattering in *dt*-*y* %-plane, Figure 2.

An analysis was done by projecting the points in Figure 2 to the *dt*-axis, one by one. The *dt*-axis projection of these numbers, Figure 3, indicates that

- the full validity of ElCo about seasons realised in 305 of 2728 circuit-groups, i.e. 11.1 % of cases; a random result would be 1/26 = 3.8 %;
- the moderate validity of ElCo about seasons by timing error of *dt* being at least two fortnights realised in 31 % of the circuit-groups, the random being 5/26 = 19.2 %, and
- measurements have to last for six fortnights (three months) in order to discover the highest fortnight of at least every second circuit-group.

The *y* %-axis projection, Figure 4, indicates that

- the average intensity error is 7.6 %, being large (14 %) for low, up to ten erlangs, and low (5 %) for high traffic, at least one hundred erlangs
- if a reliability of 90 % is aimed at, the pre-selected fortnight intensity must be exceeded by 25 %. It means corresponding overdimensioning on an average, leaving, however, every tenth intensity value underestimated.

The average intensity error in classes of timing error is presented in Figure 5. It indicates that when deviating from the perfect timing, the intensity error suddenly jumps to the rather constant high values, i.e. the highest fortnight does not stay in the neighbourhood of the earlier peak season, but appears at an arbitrary number of fortnights apart.

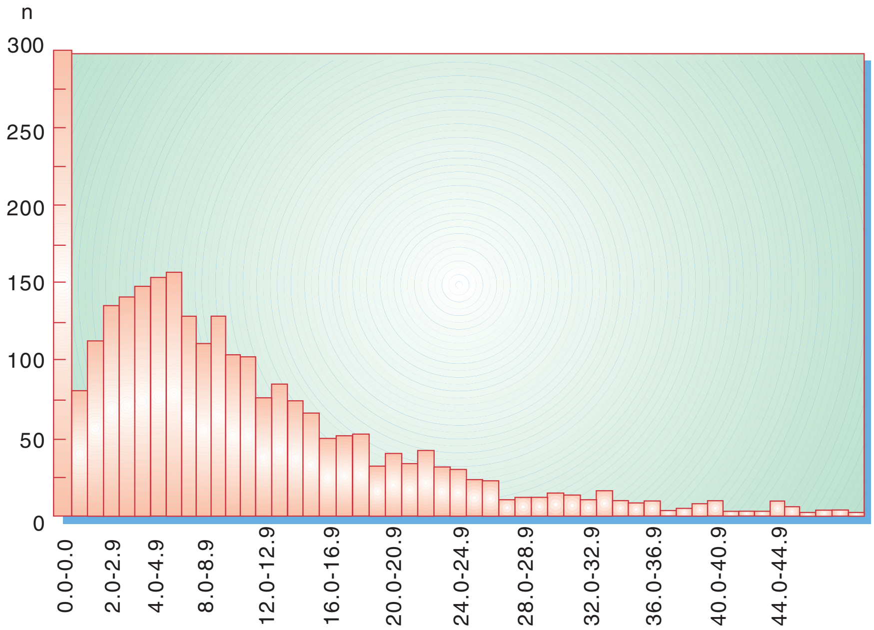


Figure 4 Distribution of intensity errors y % [5]

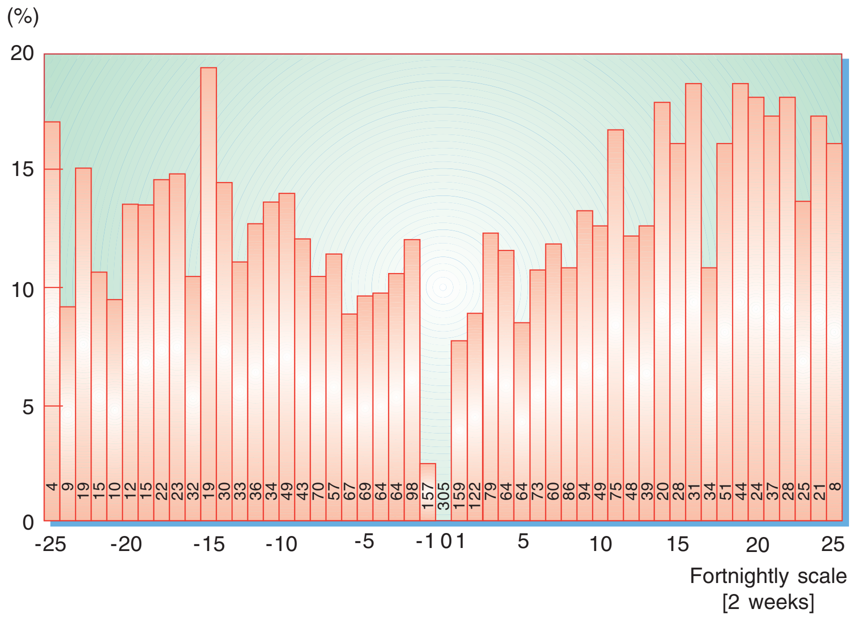


Figure 5 Average intensity errors in different timing error classes [5]

The validity of ElCo about seasons is less than 50 %. The high season thus cannot be nominated in advance, but intensities of consecutive fortnights can vary strongly. Only the low loads during summer vacations can be identified, the decrease being on an average of 30 %. Since the high load timing is not predictable, the measurement has to last for periods totalling at least six months, uninterrupted or in shorter periods.

4 The stability of the peak-hour in measurement rounds

The *ElCo about measurement rounds* is that *the peak-hour of the average day in a measurement round of a circuit-group is stable from season to season*. If the ElCo about measurement round is valid, the intensity measurements can be con-

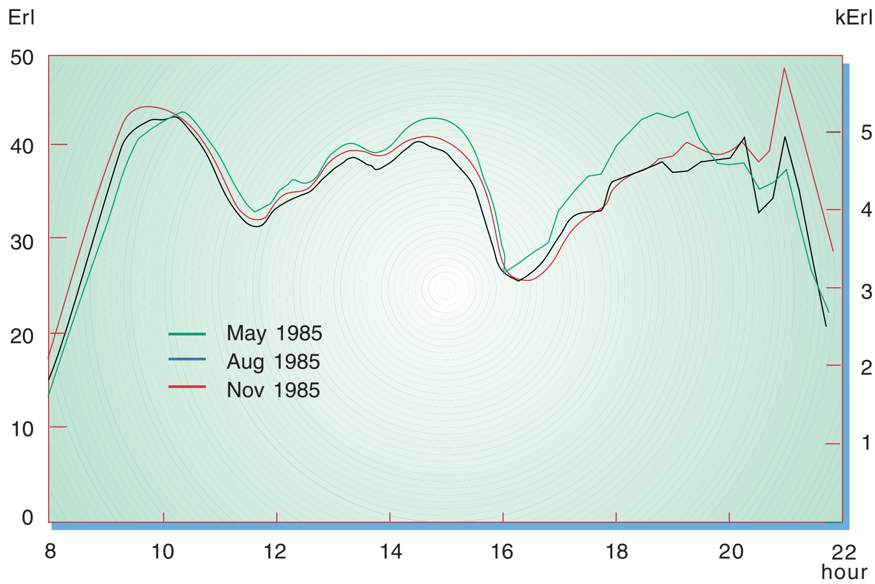


Figure 6 The daily traffic intensity profiles, the quarter-hours being averaged over ten consecutive working days, each as average of 115 circuit-groups [5]

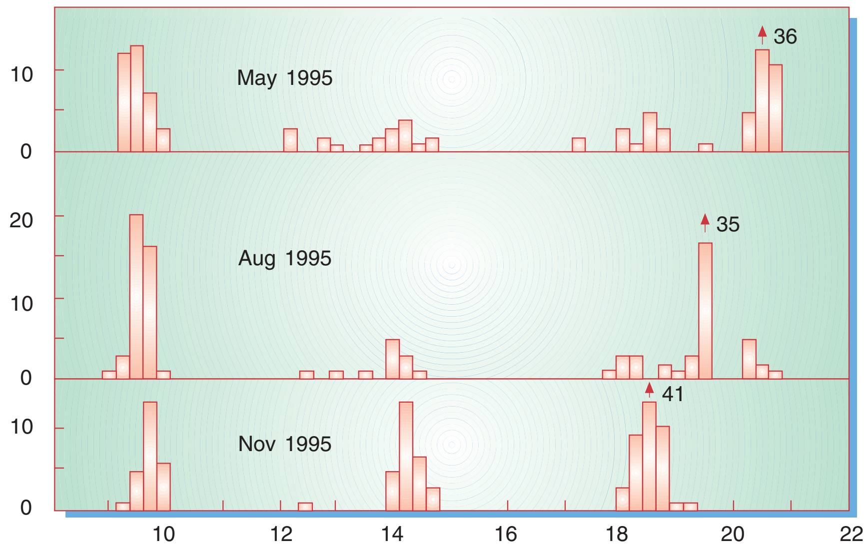


Figure 7 Division of the 115 circuit-groups into different peak-hour timings in the three fortnight measurement rounds: May, August and November 1985 [6]

centrated on these hours only, where a peak-hour has been discovered in earlier measurements.

The *study object* in Helsinki local network was the 115 circuit-groups in the Sörnäinen AKE-transit exchange. The fortnight measurement rounds in quarter-hours lasted daily from 8 a.m. to 10 p.m. The measurements were carried out in May, August and November 1985, totalling nearly 600 kEh per round. The complete study is published in [6].

The study indicated that the total profiles of the whole exchange, describing sum profiles of all circuit-groups, were quite stable, Figure 6. The morning, afternoon and evening peaks are obvious, and according to the ElCo about measurement round for the whole exchange, serving both business and domestic traffic.

When observing separate circuit-groups' peak-hours, they can be seen to have morning, afternoon or evening day-time character, Figure 7.

But when observing one circuit-group in consecutive rounds, the *peak-hour jumps* from one day-time to another between rounds in 26 % of cases. According to a study of the primary material in rounds in August and November, if measured in full hours, only in five circuit-groups did (4 %) the peak-hour keep its timing strictly from one round to another. But when taking into account the partial overlappings, too (Figure 6), the percentage of overlapping circuit-groups increased. A partial overlapping of three quarter-hours was in 24, of half-hours in 14, and of one quarter-hour in 15 circuit-groups, but as many as every second case remained without overlapping at all. Thus the average overlapping was (5 x 4 + 24 x 3 + 14 x 2 + 1 x 15) / 115 = 135 / 115 = 1.17 quarter-hours. The result is that the probability of finding the peak-hour from the earlier timing in consecutive measurement rounds is 1.17/4 = 0.29. Thus the earlier mentioned good stability of the whole exchange's traffic profiles does not help the fact that by circuit-groups there is a *bigger probability to miss than to hit the peak-hour, when the measurement is timed according a preselected time*.

5 The representativity of the average day

The *ElCo about the average day is that the peak-hour of the average day in measurement round coincides with the peak-hours of the round's days*. If the ElCo about the average day is valid, the average day's peak-hour can be used instead of the peak-hours of the days in the round.

The *study object* in Helsinki local network was the 115 circuit-groups in the Sörnäinen AKE-transit exchange. The fortnight measurement rounds in quarter-hours lasted daily from 8 a.m. to 10 p.m. The measurements were carried out in May and November 1985, totalling nearly 600 kEh per round. The complete study is published in reference [7].

The study indicated that the average day's peak-hour coincides fully with the peak-hours of zero to six of the round's days in May (zero to five in November), averaging 2.36 (1.93), of the ten possible days. But when taking into account the quarter-hourly overlappings, too (Figure

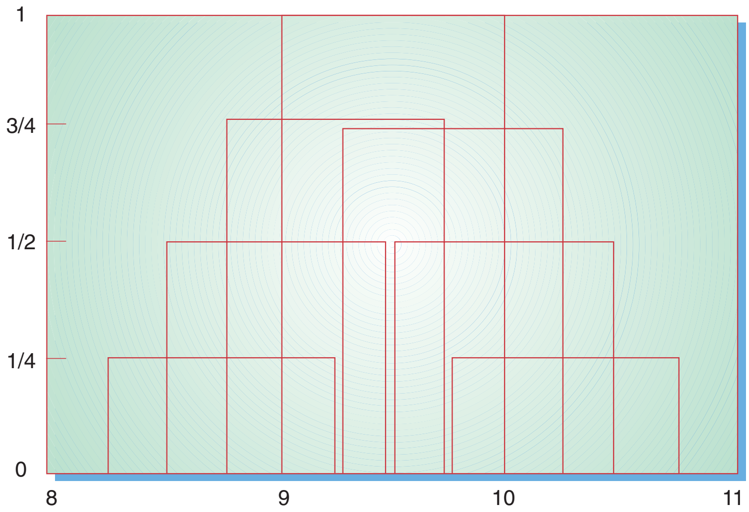


Figure 8 The weights of hours, overlapping more or less with the average day's peak-hour [7]

8), the average day's peak-hour overlaps with the peak-hours of zero to nine round's days (1 to 8.25), averaging 4.86 (4.52) days. Thus it can be said that the average day's peak-hour represents two of the round's ten days' peak-hours precisely, a little for nearly three further days, but not at all in the remaining over five days.

The measurement round's individual and average days' profiles are illustrated for two different circuit-groups, Figures 9 and 10.

It is seen that in tightly dimensioned first-choice circuit-group the daily variations are limited by the number of circuits (Figure 9), and the average day's peak-hour is not far from the individual peaks. But the last-choice circuits, especially if they are loosely dimensioned, can have large daily variations, substantially exceeding the average day's peak values (Figure 10). Then the days' peaks can hardly be expected by using the average day. The averaging over the round's days just destroys the valuable information about peak values. In such a case the average day does not give any advice for quality setting, dimensioning or optimising. – Here is an example showing that the dimensioning hour definition is more important than the distributions during that hour.

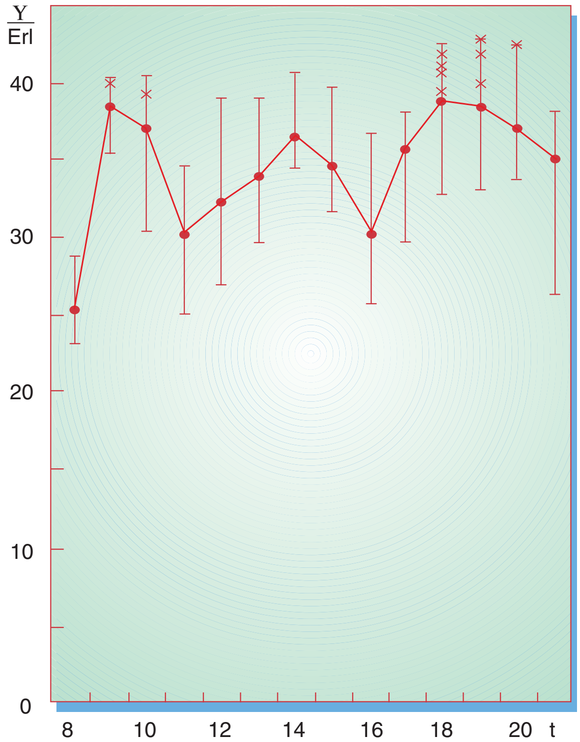
The validity of ElCo about average day is quite weak, being an exception, not the rule. In addition, the average day destroys the most important dimensioning information. Therefore, one may ask if there is any real use of the average day concept. One usage for the average day per exchange was presumably in producing the manual exchange operator's working hour schedules. The circuit-group's average day's profile might be used in network management to schedule daily reroutings in those few cases only, where all the week day profiles are similar. But there is no use for the average day in normal circuit-group dimensioning.

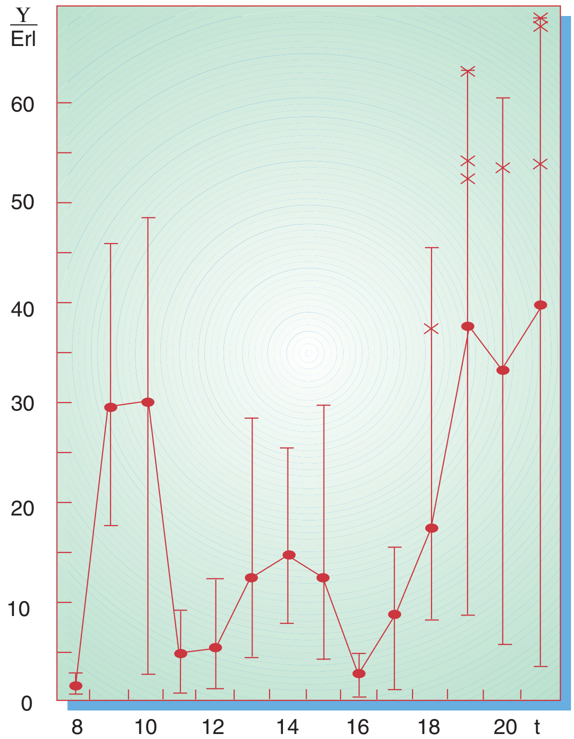
6 The ideality of traffic during the peak-hour

The *ElCo about peak-hour's model* is that *during the days peak-hour the offered traffic is in a statistical equilibrium and its distribution is ideal (Poissonian).* If the ElCo about peak-hour's model is valid, the simple mathematical models about quality relations apply. The most favoured model is the Erlang B-formula with its derivatives in waiting and overflow technologies.

The *study object* in Helsinki local network was the Sörnäinen AKE-transit exchange having 120 circuit-groups. The fortnight measurement rounds in quarter-hours lasted daily from 8 a.m. to 10 p.m. The measurements were carried out in May 1988, totalling 110 000 quarter-hours. The complete study is published in [7].

In order to have a reference level, a "Class 1" Poisson distribution's process-distinctive natural "basic" variation D is defined. The observed differences are





Figures 9 and 10 A typical ten days' three-top (A) or two-top (B) average intensity profile -o- with its daily variations, measured in May 1985. Peak-hours per day are marked by x. Note the limited variations in highly loaded 45 circuits first-choice circuit-group (A), and the wide variations in the loosely dimensioned 90 circuits last-choice circuit-group [7]

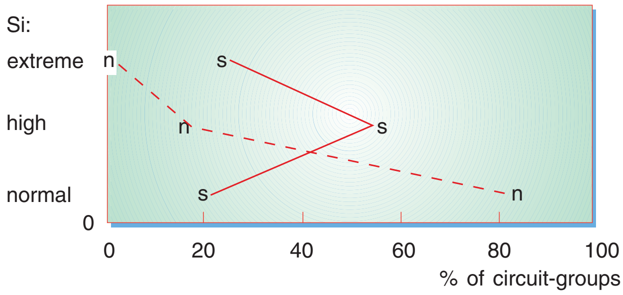


Figure 11 Distribution (s) of the observed skewness indices (Si) compared with the skewness of the normal distribution function (n) [7]

then compared with the *basic variations D*. $D = (2t_m Y/T)^{1/2}$, where t_m = average holding time, Y = measured intensity in the peak-hour, T = integration period, or the read-out period.

The peak-hour's *internal skewness* was studied by comparing its half-hour intensities. It was seen that only in 20 % of the circuit-groups the skewness was normal, i.e. in measures of the basic variation (Figure 11). The skewness was remarkably higher in 58 % and very remarkably higher in 22 % of the circuit-groups, when according to the Poisson-process, the corresponding percentages were 83.8, 15.7 and 0.5. The variations are thus clearly higher than ideal. This can be understood so that the offered intensity is not constant but changing during the peak-hour. Thus, the congestions are not divided evenly over the full hour, but concentrated to one of the half-hours. Why dimension during the full hour when the process, relevant to the quality, happens during one half-hour only? – This alone is a reason for reconsidering the modelling.

In lack of more detailed data, the peak-hour's *internal distribution* was studied as the quarter-hours' relative range W, i.e. the difference of the highest and lowest quarter-hour intensity in relation to the basic variation. The observable range W depends both on the offered traffic's variations, and the number of circuits if it is dimensioned to limit the variations. In order to distinguish these two reasons, the circuit-groups were classified according to their loading. The Christensen formula's goodness factor h is here used for load classification:

$$h = (n - Y) / Y^{1/2},$$

where h = the load index, n = number of circuits, Y = average of the round's days' peak-hour intensities.

Here the dimensioning is called tight, when $h \leq 1.5$, but normal or loose when $h \geq 2$. Tight dimensioning is used typically in overflowing high-congestion first-choice circuit-groups, whereas loose dimensioning is used in low-congestion last-choice, and in single circuit-groups.

According to the mathematical tables of Poissonian distribution, probabilities 10 %, 40 %, 40 %, 10 % are reached by the range of four samples in corresponding classes having W = < 1.0, 1.0 to 1.9, 2.0 to 3.2, and > 3.2. The numbers of circuit-groups in the same classes are given in Table 1.

It indicates that the circuit-groups in all are rather well in line with the ideal ranges. However, this is only partly a consequence of ideal offered traffic, but caused by the combined effect of tight dimensioning and peaky traffic, namely

- tight dimensioning limits the variations to narrower than the Poissonian
- loose dimensioning allows the unlimited variations of the offered traffic, being generally larger than the Poissonian
- role of the circuit-group – first-choice or individual or last-choice – has minor influence on the range, as was seen separately.

Table 1 The range in measured circuit-groups and in the Poissonian distribution, depending on the dimensioning

Classes of range	<1.0	1.0 to 1.9	2.0 to 3.2	>3.2	total
Circuit-groups	19	50	26	25	120
%	6	42	22	21	100
Of them tight	19	11	0	0	30
%	63	37	0	0	100
loose	0	39	26	25	90
%	0	43	29	28	100
Poissonian %	10	40	40	10	100

The explanation of the large range in offered traffic is that the intensity is not constant during a period of one hour, but changes by more than the normal variation. One hour is thus too long a period for the assumption of equilibrium. – This result is thus reverse to the English school which has favoured the smooth models.

When the distributions in reality differ so much from the model in common use, there are two choices:

- to develop the non-Poissonian models which better describe the large variations of the hour. The overflow dimensioning methods give some experiences in this way, or
- to diminish the dependability of the model by shortening the read-out period so far that the assumption of constant intensity is valid, from one hour to e.g. a quarter-hour.

A similar phenomenon was observed in Helsinki at an early stage [1] when the observed and the calculated congestions were compared. The calculated congestions were too few when the read-out period was half-hour, less differing in 1/8, and well fitting by 1/24 hour. The only one analysed circuit-group does not give basis for wide conclusions, but offers an interesting view for the modellers.

7 The day's peaks' concentration to the peak-hour

The *ElCo about the peak-hour concentration is that the day's peaks are concentrated into the peak-hour or to its vicinity*. If the ElCo about peak-hour concentration is valid, special attention must be paid to the perfect peak-hour timing.

The ElCo about peak-hour concentration supports the pre-selection of the measurement hour.

The *study object* in Helsinki local network was the Sörnäinen AKE-transit exchange with its 120 circuit-groups. The fortnight measuring rounds lasted daily from 8 a.m. to 10 p.m. in quarter-hours. The measurements were carried out in May 1988, totalling 110,000 quarter-hours. The complete study is published in [7].

The peak loads outside the peak-hour were classified as *mild* peaks, having an intensity less than the peak-hour average plus basic variations, and as higher *strong* peaks. It was seen that

- 32 % of the circuit-groups had no outside peaks
- mild peaks were found in 34 % and strong peaks in the 44 % of circuit-groups during at least one day
- due to the peak quarter-hours aside the peak-hour, the peak-hours measured on quarter-hour basis give on average 4.0 % higher intensity than on full hour basis
- strong peaks were found in 20.3 % of days on average in last-choice circuit-groups but 1.5 % in first-choice and 8.25 % of single circuit-groups. This might be caused more by the tightness of the dimensioning than by the role of the circuit-group, but this aspect has not been studied
- the day's outside peaks were primarily not near the peak-hour, as the so-called *side-hour concept* supposes. It might work in that direction on the average day, which might be the reason why it is favoured when the peak-hour is timed on quarter-hour basis on the average day. It does, however, not help much with the uselessness of the average day.

Thus, the ElCo about peak-hour concentration can be seen to be valid in every third case only.

8 Comparison of some measurement routines

Caused by the phenomena described above, the choice of the measurement routine influences the produced intensity values. The measurement routines are *non-continuous,* scheduled according to the expected peak loads, or *continuous,* the relevant information about peaks chosen afterwards by data post-processing. Scheduling concerns the year and/or the day. The several e.g. fortnight measurement rounds can be grouped in some way over the seasons. The measurement periods of the day define the scheduled routines.

The following measurement routines, included in the ITU-T (late CCITT) recommendation E.500, are compared by their measured intensity values:

- The continuous post-selected peak-hour of the average day *TCBH* (Time-Consistent Busy Hour) on quarter-hour basis. (The name is misleading in many ways; a better one were Peak-Hour of the Average Day, PHAD)
- Average of continuously measured post-selected peak-hours *ADPH* (Average of Day's Peak-Hours) on full hour basis
- Pre-selected measurement hour *FDMH* (Fixed Daily Measurement Hour), the hour selected earlier by the TCBH-routine
- Peak-hour of the average day's pre-selected measurement hours *FDMP* (Fixed Daily Measurement Period).

The *study object* in Helsinki local network was the 115 circuit-groups in the Sörnäinen AKE-transit exchange. The measured fortnight rounds lasted daily from 8 a.m. to 10 p.m., in quarter-hours. The measurements were carried out in August and November 1985, totalling nearly 600 kEh per round. The complete study is published in [6].

When the result of TCBH-routine is used as reference (= 100 %), the other routines give values as in Figure 12, describing averages of the August and November measurements, and the typical variation limits (until 2s or 100 %) are given. The circuit-groups' intensities are combined without any weighting. The non-continuous measurements were scheduled according to the peak-hour of the earlier measurement of each individual circuit-group, or for the exchange commonly.

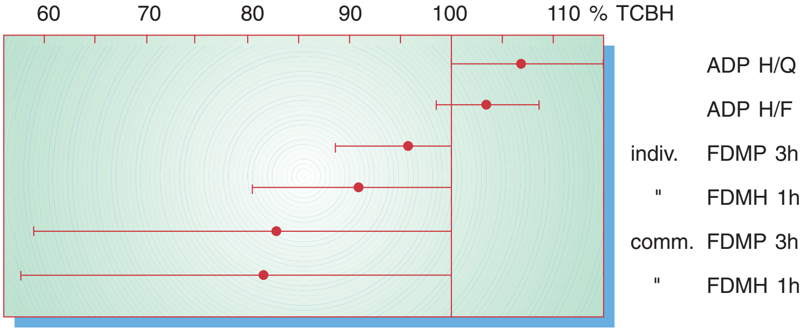


Figure 12 Intensity values, measured in different continuous and noncontinuous routines, as percentages of the TCBH intensities -o-, with the confidence limits [6]. The ADPH is presented in two versions, namely measured by quarter-hour /Q or by full hour /F basis

The continuous TCBH- and ADPH-routines give thus very similar results, and the same dimensioning rules and tables have been recommended for both of them [2]. But the TCBH is not usable if the day profiles in the round differ remarkably, as is the rule e.g. by weekend peaks and last-choice overflow circuit-groups. The scheduled routines can occasionally give results high enough, but mostly their average is clearly below the real load, 10 to 20 %. The occasional underestimate can be up to 20 or 40 %, or even more. There is no means of identifying afterwards which ones of the circuit-groups were underestimated. – In this comparison the circuit-groups having tight or loose dimensioning were not distinguished; the results present their total. Presumably a loose dimensioning causes more differences of the intensities, produced by different routines, than a tight dimensioning.

9 Measurements of overflow cluster for optimising

A single circuit-group is dimensioned according to its service goal. But in the overflow clusters the aim has not only service as its goal, but an *optimum of the total cost*, covering investments and usage of all first and last choice circuits of the cluster. Compared with a radial network structure, the direct route (e.g. the first choice circuit-group) causes a cost increase which must be covered by diminishing the radial (e.g. the last-choice) circuit-group. The calculation methods to discover the optimum size of the circuit-groups are mainly developed under the assumption of the ElCos above, e.g. the coincident peak-hours of different traffic flows in a cluster, even assuming corresponding regularities not only in the individual traffic flows, but synchronously in all flows belonging to the cluster.

If the peak-hours are non-coincident, the advantage gained from diminishing the last-choice circuit is missed more or less. In other words, the optimal size of the first-choice circuit-group diminishes with growing non-coincidence of peak-hours. In order to control the dimensioning by real traffic relations, the measurement routines need different qualifications from those of the single circuit-groups.

Dimensioning of the *last-choice circuit-group* defines the service quality of the cluster. Due to overflowing, the traffic distribution during the peak-hour is more peaky in the last-choice circuit-group than traffic offered to the first-choice circuit-group. This should be made clear by means of measurement routines and modelling.

Dimensioning of the *first-choice circuit-groups* defines the overflow and thus, the economy of the cluster. Starting from balanced cost, caused by carrying one erlang alternatively by last-choice or first-choice circuits, a Moe-minded thumb-rule for dimensioning the first-choice circuit-group is created:

$$F(A)_{\mathrm{n}} = \varepsilon H,$$

where $F(A)_n$ = carried traffic caused by the n'th first-choice circuit, ε = cost of one first-choice circuit in units of the cost per one last-choice circuit, and H = traffic per last-choice circuit. Example: $\varepsilon = .5$, $H = .6$, and $F(A)_n = .3$ gives the needed relations between A and n. – The formula is valid as long as the augmented first-choice circuit causes gain in the last-choice circuitry in question. This is not realised in cases where, in spite of a noticeable overflow, some last-choice circuits have no load due to minor traffic offered to them from other sources. This relation should be clarified by means of measuring and modelling.

Peak intensities in last-choice circuit-groups can be clarified in two alternative ways, namely by calculating from peak-hour average by model or by measuring a short peak period to be calculated further.

Calculating the peaks by some equivalence rules and thereby the congestion, is based on offered traffic having Poissonian distribution. Such methods have been developed by Wilkinson, Bretschneider, Rapp and others. The method is mathematically clear. In practice, it is ambivalent when the offered traffic is more peaky (see above) and the real distribution cannot be studied case by case. The method has been used in pre-selected measurement hours, too. When omitting the peaks outside the peak-hour and measurement period, it gives an indefinite picture of the service quality, and is not usable in network structure optimisation.

Measuring the peaks during a period shorter than one hour reveals higher peaks, during which the distribution approaches the Poissonian. Calculating congestion leads to a better result. The method is mathematically less defined, but yields results with better applicability through its diminished sensibility to the model of offered traffic. Consequently, the peaks outside peak-hours and the daily variations are included thereby.

In *dimensioning the first-choice* circuit-group, the load during peak-hour of the last-choice circuit-group is of most importance to optimising; the first-choice circuit-groups' own peak-hours do not influence the optimum.

Measurement routine of first-choice circuit-groups can be created in two ways in order to minimise the collected data, namely by a synchronised or a softened routine. Both of them yield the intensity for the above mentioned thumb-rule equation.

In a *synchronised routine,* the measurement of the last-choice traffic is activated continuously, but the first-choice is on stand-by. The last-choice meter discovers and reports the days' peak hour intensities, but the first-choice meters are activated to report the coincident hour's intensity, not their own peak-hours. Such a routine works in cases where the successive last-choice circuit-groups' peak-hours can be treated as one, alternatively due to their peak-hours actually being coincident, or while only one of the circuit-groups is relevant (the others being e.g. overdimensioned). The measurement becomes complicated when the cluster consists of several first-choice circuits with non-coincident peak-hours, and several last-choice circuit-groups are economically relevant. Such a many-conditioned synchronising is hardly realisable, not even by post-processing the complete profile data. – The synchronised measurement can be regarded as theoretically sound, but inadequate in practice for the complicated clusters.

In a *softened routine* the measurements of first-choice traffic aims at an intensity value which at least will be reached during the last-choice peak-hour with considerable certainty. Instead of the first-choice peak-hour, it is aimed at a lower value fitting with the last-choice peak timing. One approximation of the searched intensity might be an average integrated over a period of a few hours. Its length depends on the variations of the day's profiles and the seasonal variations, and it should be fixed after wide studies of real cluster traffic, non-existent so far. A good guess of two to four hour periods could be a starting point for the studies.

10 Comparison of measurement routines in overflow clusters

The usability of some measurement routines were studied in some existing overflow clusters. The *study object* in Helsinki local network was the 115 circuit-groups in the Sörnäinen AKE-transit exchange in May 1985, as described above. The complete study is published in references [7, 8].

Fifty-eight circuit-groups among the 115 were combined as overflow clusters. Half of them were ignored in this study, because the first-choice circuit-groups were too loosely dimensioned to have overflow of high congestion, but only of a low congestion. The planners had apparently overestimated the first-choice traffic during the last-choice peak-hour. Thus, a marked overflow was available in three clusters only. Among them there was a small cluster with simultaneous peaks in different flows, and a big one with non-coincident peak-hours. All the 17 circuit-groups in question were measured in parallel using the three continuous routines, namely TCBH, ADPH and ADSI. ADSI (Average of Daily Synchronised measured Intensities) routine measures the first-choice circuit-groups continuously, but is automatically activated to report only during the peak-hour in the common last-choice circuit-group, revealed by its continuous measurement.

It was observed that

- if the peaks of different traffic flows are simultaneous, as in one small cluster under study, all the continuous routines TCBH, ADPH and ADSI are equivalent
- if the peak-hours of different traffic flows are non-coincident, as in one small and the big cluster under study, the day's profiles in the last-choice circuit-groups are very unstable, Figure 10. In that case the average day has compensated the variations of importance, and the TCBH gives too low values
- the ADPH reports about the peak-hour intensities in all circuit-groups, but does not reveal the secular relations between traffic peaks of last- and first-choice circuit-groups
- in the big cluster, when keeping the overflow, being calculated from ADSI-measured intensities as a basis, the TCBH gave intensities 40 to 100 % higher, and the ADPH 70 to 200 % higher. In that case the ADPH in one hour periods is less suitable for the first-choice measurements.

The overflow measurements are presented here as examples only. A wide study would be needed to define the traffical circumstances in overflow clusters.

Several calculation methods for overflow-cluster optimising have been developed, concerning the even complex constructed models during peak-hour. Under the assumption of coincident peak-hours, many high-congestion circuit-groups seem to be economical and to cover their initial costs. But because the peak-hours of different traffic flows in fact are non-coincident, many of the direct routes cause extra cost, which exceed the advantage gained by diminishing radial circuits [3, 4].

11 Normal and high load of the year

According to ITU-T Recommendation E.500 the annual intensity can be measured by a routine producing "normal load" or "high load". The use of these values differs in dimensioning. The daily measurement hour is pre-selected to be the peak-hour of the average day's profile, which is assumed to keep its position over the year. The *normal load* is the average of 30 highest days' intensities measured during a pre-selected hour, whereas the *high load* is created as five highest days' average. – Dimensioning hour intensities of that kind are complicated to collect, they are sure to miss many peaks, and this routine is not usable in forecasting by time series. This year's highest days' routine in Rec. E.500 dates from the 1960s, but is not known to have been used as a routine by any network operator. Some organisations have preferred to keep it, anyhow, as a basic measurement standard for normal and high load.

The intensities measured using the popular ten consecutive day's measurements in TCBH-, ADPH- or scheduled routines are given in Rec. E.500 as parallels for the normal load. However, no way was given in which to measure the *high load with the ten consecutive days'* routines. This was looked at in a study in Helsinki local network [11], based on the 2321 circuit-groups' night-and-day AUTRAX data over the whole year 1990, totalling over 100 MEh. The study indicated that

- the TCBH- and ADPH-routines give intensities equalling the 30 highest days = normal load, when the consecutive ten measurement days period is selected in advance
- the ADPH-routine gives intensity equalling the 5 highest days = high load, when the ten days are selected afterwards to be the highest of the year.

According to the Rec. E.5xx series recommendations, the nominal congestion is 1 % when dimensioning circuit-groups by intensities of normal load, but 7 % by the high load intensities. It means that in the exchange systems with continuous measurement routines, circuits can be saved compared to the systems with only by demand connected, or scheduled measurement routines, because the augmentations in the former are better focusable according to the peak loads.

12 Conclusion

The studies about Elementary Conceptions and usability of some measurement routines showed that among the annual, weekly or daily *variations of traffic* one can find regularity, exceeding the pure random. This is, however, so weak that it does not support concentration of measurements to short pre-selected periods. Therefore, measurements have to be continuous and the peak values are chosen among the data, preferably immediately. – If all the traffic data is stored in files of per call tickets, the peak loads can be reconstructed afterwards, too. The delay worsens the freshness of data and speed of the operational reactions.

The statements, given above, have been tested partly by wide, partly by limited *measurements.* The measurements above have been reported and discussed by specialists over the years. The measured results well describe the situation in the Helsinki telephone network in the 1980s. Concerning telephony, they might still be valid there. In Helsinki, two- or three-peaked daily profiles are often caused by the mixture of domestic and business traffic. This effective use of circuits is a consequence of communal master plans. The validity of the statements is not verified in other cities. – But in intercontinental east-west traffic with only a few common office hours the profiles hardly fit with the ones described above.

Dimensioning of the circuit-groups has been based on *one hour* since the beginning of the century. The long integration time has been motivated in order to minimise handling of the data because the equipment was clumsy and data handling manual. And the disadvantage of uncertainties was not too disturbing in traditional telephony with manual switching and marked waiting times. But now one hour is too long for the assumption of a statistical equilibrium, the real peaks being higher than the common model presumes. In the near future the voice traffic might require integration periods of half- to quarter-hour. – Concerning short occupation time processes, like in data or signalling traffic and processors loads, clearly shorter read-out periods are needed. The CCITT Recommendation E.500 has thus been augmented by a formula [10] which reduces the read-out periods, measured by non-one-hour periods, to dimensioning on one hour basis.

In the overflow-cluster *last-choice* circuit-group, defining the service quality has to be dimensioned by its peak intensities, preferably in quarter-hour readouts. For such short peaks during the year, nominal congestion of several percent can be allowed. The second moment calculations, combined with the number of the first-choice circuits, are hardly needed if the peaks are measured directly.

For dimensioning of the *first-choice* circuit-group which defines the cluster's economy, its peak-hour is not usable. A lower than the peak value should be found which coincides with sufficient probability with the last-choice circuit-groups peak-hour. An ADPH-routine, with two to four hours read-out, can be proper. The dimensioning can then follow Moe's principle.

Many of the overflow networks are very complicated, with lots of alternatives. The complexity of the network may be limited by the risk of jeopardising its manageability, the traffic control and the means of measuring. If these are exceeded, the complication can become problematic.

One may ask if these observations and conclusions, mainly dating from the 1980s, are *pertinent* in the future network services: now the components are cheap and capacity costs almost nothing. A few comments will be given.

Service quality of a network depends on some bottlenecks in the network. These impair the service when the load limit is reached. The bottlenecks are probably not in transmission routes any more, but in signalling, data transmission channels or processors, getting blocked first. These limits must be identified by the operator, who has to know how near the calculated limits the actual loads are. There, the peak-hour or other dimensioning period intensities with their models will be needed for capacity calculations, even in the future. Some of the load variations follow these, as described above, some are of new types. Measurements clear the traffical facts – models just help in thinking. The augmentations can then be prepared in time. Otherwise, the operator will be in for a surprise.

Network economy becomes more complicated when, instead of circuit-groups, the dimensioning objects are clusters, networks, and groups of competing operators' networks, serving clients of various services. This higher level of traffic dimensioning and optimising is not easy to manage without knowledge of the simple components' traffical relations. Or what is even worse, if the complicated network is still controlled on the basis of obsolete elementary conceptions.

Traffic modelling traditionally concentrates on Poissonian distribution inside the hour and regular variations between the hours, days and seasons. The above described measurements do not support this concept: traffic variations usually exceed the expected. Rather some grade of self-similarity, known in mathematics of fractions, can be found, which opens ways to very different thinking in the future.

13 Acknowledgements

The above described measurement have been made during several years, partly to answer some questions raised by the CCITT Study Group 2, partly arisen from local interest. I owe my warm thanks to the following specialists and colleagues at Helsinki Telephone Company: Dr. Pekka Lehtinen programmed the AUTRAX equipment to produce several traffic data simultaneously and continuously, and analysed traffic variations. MSc Matti Mäkelä and BSc Kari Laaksonen with their staff carried out the traffic measurements in practice. MSc Yrjö Viinikka of Oy Comptel Ab took care of the data handling.

14 Documentation

The above referred studies about optimising and the traffic measurements in Helsinki, have been published in full in the following papers:

1 Ahlstedt, B V M. Load variance and blocking in automatic telephone traffic: load duration, a basis for dimensioning : blocking duration, a yardstick for traffic quality. I: *5th international teletraffic congress, ITC 5,* 524–530, New York, 1967.

2 Lehtinen, P I. Extreme value control and additive seasonal moving average model for the evaluation of daily hour traffic data. I: *9th international teletraffic congress, ITC 9,* Torremolinos, 1979.

3 Parviala, A. The suboptimal alternate routing practice with non-coincident busy hours. I: *10th international teletraffic congress, ITC 10,* Montreal, 1983.

4 Parviala, A. The dimensioning of alternate routing by a simple algorithm, taking into account the non-ideal parameters. I: *5th Nordic teletraffic seminar, NTS 5,* Trondheim, 1984.

5 Parviala, A. The stability of telephone traffic intensity profiles and its influence on measurement schedules and dimensioning. I: *11th international teletraffic congress, ITC 11,* Kyoto, 1985.

6 Parviala, A. The stability of daily telephone traffic intensity profiles and its influence on measurement routines. I: *6th Nordic teletraffic seminar, NTS 6,* Copenhagen, 1986, and Sähkö 60, 26–30, 1987.

7 Parviala, A. Peak-hour and other peaks of day. I: *8th Nordic teletraffic seminar, NTS 8,* Espoo, 1987.

8 Parviala, A. The stability of the daily intensity profiles and its influence on the choice of measurement routines in telephone traffic in single low-congestion circuit-groups or overflow clusters. I: *12th international teletraffic congress, ITC 12,* Torino, 1988.

9 Parviala, A. The reality of busy hour in history and in measurements. I: *9th Nordic teletraffic seminar, NTS 9,* Kjeller, 1990, and I: *13th international teletraffic congress, ITC13,* Copenhagen, 1991, WS History.

10 Parviala, A. The year's highest vs. consecutive days. I: *10th Nordic teletraffic seminar, NTS 10,* Aarhus, 1992.

11 Parviala, A. Traffic reference period. I: *11th Nordic teletraffic seminar, NTS 11,* Stockholm, 1993, and I: *14th international teletraffic congress, ITC 14,* Antibes, 1994. Spec. Traffic Measurements.

Point-to-point losses in hierarchical alternative routing

BY BENGT WALLSTRÖM

Abstract

In a previous paper [9] the joint state distributions of carried and overflowing partial traffics were considered, suggesting possibilities of point-to-point blocking calculations in alternative routing networks. The present paper will show how a simple extension of Wilkinson's Equivalent Random Theory (ERT) can be used for that purpose in a hierarchical network.

1 Introduction

To recall some basic elements of overflow traffic theory, let us consider the scheme (Figure 1) studied by Wilkinson in his *Theories for toll traffic engineering in the USA* [10].

The left hand arrows represent traffic streams ($A_{01}, A_{02}, ..., A_{0g}$) that are primarily offered to their own direct links having each $n_{01}, n_{02}, ..., n_{0g}$ channels. Whenever a new call finds its own link fully occupied it will search for a free channel among those k_0 available on the common link. A_0 is to denote a traffic stream offered directly to the common link. The structure is readily recognised as a part of a typical alternative routing pattern, specially useful between a lower and the next higher network level. Here several overflow streams may be combined and offered to a common 'upward link' in the network. Of course there is a similar structure of downward links where we pass from higher to lower network levels. This implies repeated splitting of traffic streams.

Considering Wilkinson's theory and similar approximate methods it seems that the combination of overflow parcels on upward links is quite well described while the splitting performance on downward links could perhaps be better done. The main objective of this study is an attempt to calculate improved characteristics of typical split traffics in alternative routing networks. In section 2 we recall the Equivalent Random Theory (ERT) to state its main assumptions and results. In section 3 an approach to split traffics calculation is presented with moment formula that can be computed from results given in ERT.

2 ERT – Equivalent Random Theory

The ERT was initially designed to attain optimal dimensioning of hierarchical alternative routing networks. In the following discussion we retain the assumption of such a routing structure but note that ERT has also proved more generally useful.

The routing scheme of Figure 1 may depict traffic streams in a network where a group of basic switching centres are connected to a primary transit centre. One of the basic nodes is given direct access to links to g equivalent nodes. Traffics overflowing direct links are served together on a common link to the transit centre. Thus alternative routing through the transit node may be possible.

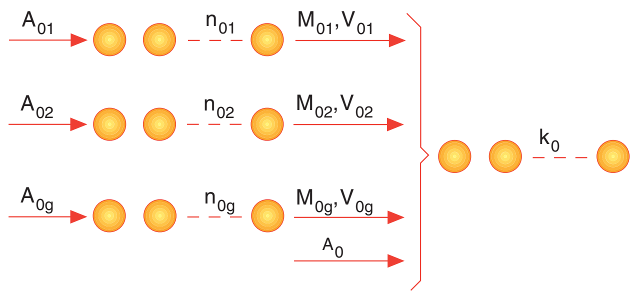


Figure 1 Basic overflow scheme of an alternative routing network

The network links of Figure 2 are readily recognised in Figure 1 but in addition there are shown downward links from the transit node to each of g basic nodes. Also we may identify g other similar patterns, each defined by g links to basic nodes, one upward transit link and g downward transit links.

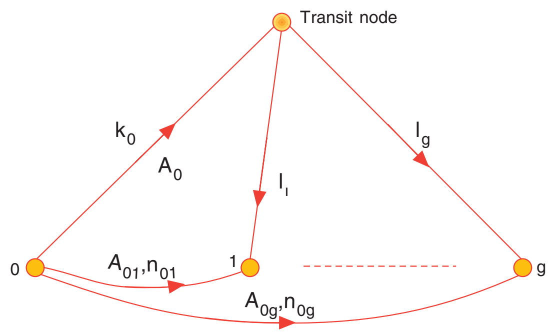


Figure 2 Direct and alternative routes from one basic node to g other ones

Let us calculate the number of transit channels, k_0, of Figure 1 according to ERT.

- Traffics denoted $A_{01}, A_{02}, ..., A_{0g}$ and A_0 have all the Poisson-traffic character. They are generated by independent Poisson-arrival processes and share a common exponential service time distribution.
- The intensity, A, of a Poisson-traffic is defined as the mean number of calls in progress in an infinite group of channels and is referred to as offered traffic. The probability of blocking in the finite group of n channels is obtained from the Erlang loss formula

$$E_n(A) = \frac{\frac{A^n}{n!}}{1 + A + \ldots + \frac{A^n}{n!}}$$

- Traffics overflowing from the direct links $n_{01}, n_{02}, ..., n_{0g}$ to the transit link k_0 will be characterised by two descriptive parameters, the mean and the variance, to be defined below. This is done to take the typical peaked variations of overflow traffics into account. The traffic A_0 is offered directly to the common transit link k_0.

The ERT definition of overflow traffic is based on early results by Kosten [3] considering Poisson-traffic *(A)* offered a fully available primary group *(n)* with blocked calls given service in an infinite overflow group (Figure 3).

The traffic process in a Kosten system has the stationary state distribution

$P(i,j) = P\{I = i, J = j\}$

where I and J are stochastic variables denoting the number of occupied primary channels ($0 \le I \le n$) and the number of occupied secondary channels ($0 \le J \le \infty$) respectively. The formula of $P(i,j)$ is rather complicated but the lower moments of overflow traffic are quite simple and were given by Wilkinson as the mean

$$M = \sum_{i=0}^{n} \sum_{j=0}^{\infty} jP(i,j) = AE_n(A)$$

and the variance

$$V = \sum_{i=0}^{n} \sum_{j=0}^{\infty} j(j-1)P(i,j) + M - M^2$$

$$= M\left(1 - M + \frac{A}{n+1-A+M}\right)$$

Similarly as for Poisson-traffic we note that overflow traffic is defined for an infinite group and may likewise be named ‘offered traffic’. We can now calculate the means M_{01}, M_{02}, ..., M_{0g} and variances V_{01}, V_{02}, ..., V_{0g} overflowing from direct links using Wilkinson’s formula. Since the input streams A_{01}, A_{02}, ..., A_{0g} are independent Poisson traffic processes we can conclude that the individual overflow processes, too, are independent. Thus the total traffic offered to the upward transit link will have the mean

$$M_0 = M_{01} + M_{02} + ... + M_{0g} + A_0$$

and the variance

$$V_0 = V_{01} + V_{02} + ... + V_{0g} + A_0$$

Our next step is to calculate the traffic lost in link k_0 given that the offered traffic has the mean M_0 and the variance V_0.

This is done smartly be replacing the g direct links to basic nodes by a single imaginary link of n channels offered a Poisson-traffic, A, as shown in Figure 4.

Thus we calculate A and n from the Wilkinson formula given $M = M_0$ and $V = V_0$ and we can use the same formulae to estimate the moments $\mathcal{M}_0$ and $\mathcal{V}_0$ of overflow traffic from the link k_0. Especially we may estimate the average loss in link k_0 as

$$B_0 = \mathcal{M}_0/M_0 = \frac{AE_{n+k_0}(A)}{AE_n(A)}$$

$$= \frac{E_{n+k_0}(A)}{E_n(A)}$$

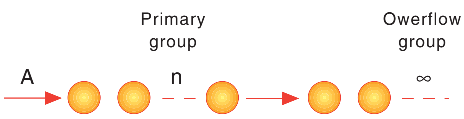


Figure 3 ‘The Kosten system’

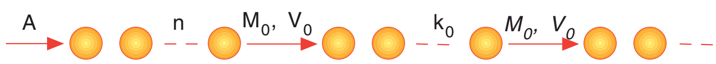


Figure 4 The equivalent random principle

For the traffic A_0 offered directly to the transit link a somewhat better loss estimate can be obtained as the time congestion, E_0, encountered in that link [1]. The results are approximate, of course, but are generally considered rather acceptable for dimensioning purposes. Though we have omitted the network outside a single transit centre and its basic nodes, this should not hide any major difficulties as long as we are contented with average losses of combined traffic streams on upward links. In the sequel, however, we will also attempt to calculate individual losses from one basic node to another.

3 Split traffics calculation

A natural approach to our problem of individual loss calculations would be to use the ERT concept shown in Figure 4, where the imaginary poissonian traffic intensity A is split as follows:

$$A = A_0 + A_1 + ... + A_g$$

$$A_i = M_{0i}/E_n(A),\ i = 1, 2, ..., g$$

Thus we have preserved the individual overflow means from direct links. This will hopefully lead to useful results also when combining split overflow traffics on downward transit links. The formula to be used are obtained for the overflow system of Figure 5 and may be found in the paper *Joint state distributions of partial traffics* [9]. They give estimates of the means $\mathcal{M}_{0i}$ and variance $\mathcal{V}_{0i}$ of split overflow traffics behind the (imaginary) group $n + k_0$ offered g independent Poisson-processes defined above. The results are

$$\mathcal{M}_{0i} = \alpha_i \mathcal{M}_0$$

$$\frac{\mathcal{V}_{0i}}{\mathcal{M}_{0i}} - 1 = \alpha_i \left(\frac{\mathcal{V}_0}{\mathcal{M}_0} - 1 \right)$$

$$\alpha_i = \frac{A_i}{A}$$

$$i = 1, 2, ..., g$$

$$\mathcal{M}_0 = AE_n + k_0(A)$$

$$\mathcal{V}_0 = \mathcal{M}_0 \left(1 - \mathcal{M}_0 + \frac{A}{n + k_0 - A + \mathcal{M}_0} \right)$$

It should be noted that the overflow processes behind a transit link are dependent. Thus we did not obtain $\mathcal{V}_0$ as a sum of individual variances. Instead we found ‘variance/mean - 1’ to be an additive characteristic of the partial overflow traffics.

The individual blocking probabilities in link k_0 are obtained as

$$B'_{0i} = \frac{\mathcal{M}_{0i}}{\mathcal{M}_{0i}} = \frac{A_i}{A} \cdot AE_{n+k_0}(A)/A_iE_n(A)$$

$$= \frac{E_{n+k_0}(A)}{E_n(A)}$$

a result that is certainly approximate and will be discussed below. Note that individual variances $\mathcal{V}_{0i}$ will not be used in our simple routing structure with low loss transit links.

Considering Figure 2 we could identify $g + 1$ link patterns, each showing g links to basic nodes, one upward transit link and g downward transit links. Let us focus on the downward link l_1, a group of l_1 channels. Among the overflow traffics offered to k_0 the only one that can reach l_1 is (M_{01}, V_{01}). Similarly, it is seen that among traffics offered to k_2 the link l_1 can only be reached by (M_{21}, V_{21}), and so on. The total traffic that can have access to l_1 may be written as the mean

$$M_1 = M_{01} + M_{21} + M_{31} + ... + M_{g1} + A_1$$

and variance

$$V_1 = V_{01} + V_{21} + V_{31} + ... + V_{g1} + A_1$$

Let us assume for simplicity that upward links, k_0, k_2, ..., k_g are all low loss links that will allow almost all corresponding overflow calls to reach link l_1. M_1 and V_1 define just a new combination of traffics similar to the one offered to the upward transit link k_0. Clearly the downward link l_1 can be treated in the same way by ERT. Thus for the simple case of an isolated network with one transit node and with low loss transit links we may estimate the loss from node 0 to node 1 as

$$B_{01} = E_{n01}(A_{01})(B'_{01} + B''_{01})$$

where B'_{01} and B''_{01} are the individual losses in the upward (k_0) and downward (l_1) links, respectively.

4 Conclusion

The methods suggested above for estimating point-to-point losses in hierarchical alternative routing networks can be judged only be extensive simulations. At present, however, we may see some general consequences of results obtained so far:

- Individual losses in the same transit route do not differ. This is clearly an approximation that follows from the simple splitting rule applied above. It should be more important, however, to be able to detect varying losses between different basic nodes and between subgroups of nodes. The Split Traffics Calculation may be a useful method to attain this.
- Though the network considered above is a very simple one it seems that point-to-point losses can be estimated by similar means for several 'nice' structures of hierarchical routing. It must be remembered, however, that the last choice route (the backbone route) should consist of low loss links only.
- From the viewpoint of point-to-point losses there are certainly many 'not so nice' topologies and routing principles that remain to be studied. Non-hierarchical routing is indeed an area of intriguing problems of that kind. Possibly the more general results given in [9] may be of some help.

Acknowledgements

I am most grateful to Prof. A. Myskja of the University of Trondheim for putting the question leading to the study [9] and then supplying ideas for the present paper. Also I would like to thank my colleagues Prof. U. Körner and Dr. L. Reneby of the Lund Institute of Technology for helpful discussions and proof reading.

References

1 Brockmeyer, E. The simple overflow problem in the theory of telephone traffic, (in Danish). *Teleteknik,* 5(4), 1954.

2 Kelly, F P. *Reversibility and stochastic networks.* John Wiley & Sons, 1979.

3 Kosten, L. Über Sperrungswahrscheinlichkeiten bei Staffelschaltungen. *Elektr. Nachr.-Techn.,* 14(1), 1937.

4 Körner, U. *Congestion control in packet switching computer communication networks,* (thesis). Department of Communication Systems, Lund Institute of Technology, Sweden. Lund, 1982.

5 Myskja, A. *A study of telephone traffic arrival processes with focus on non-stationary cases,* (thesis). The University of Trondheim, 1981.

6 Reneby, L. Service protection and overflow in circuit switched network, (thesis). *Technical report*, 107, 1991. Department of Communication Systems, Lund Institute of Technology, Sweden.

7 Wallström, B. Congestion studies in telephone systems with overflow facilities, (thesis). *Ericsson Technics,* 22(3), 1976.

8 Wallström, B, Reneby, L. On individual losses in overflow systems. I: *9th international teletraffic congress, ITC 9,* Torremolinos, 1979.

9 Wallström, B. Joint state distributions of partial traffics. I: *10th Nordic teletraffic seminar, NTS 10,* Århus, 1992.

10 Wilkinson, R I. Theories for toll traffic engineering in the USA. *Bell system technical journal,* 35(2), 421–514, 1956.

On overload control of SPC-systems

BY ULF KÖRNER

1 Introduction

Communications systems of tomorrow, handling a variety of broad- and narrow-band services, will employ switches with large switching capacities and very powerful control systems. However, no switch, and especially no control system, whatever its capacity, will be able to meet all demands that may arise momentarily. High utilisation of the switch in combination with a variety of new services with different requirements on times for call-set-up, on access-priorities under periods of overload, etc., will definitely call for very effective mechanisms for overload control. The real time capacity of the control system is in many cases the ultimate limiting resource of the switch. This precious resource has to be used, not least under periods of overload, for the most vital tasks, in order for the switch to sustain the overload. Especially, no resources should be spent on call-handling tasks that do not lead to fulfilled calls.

The primary task for the overload control mechanism is to keep the effective throughput high also under periods of overload. (See Figure 1.) In [6] it is said that *"the most obvious symptom of an overloaded SPC-system running telephony applications is the sudden occurrence of long delays in providing dial tone. While uncompleted calls occur naturally even at light loads, delay of dial tone increases their likelihood considerably"*. Field trials in the U.S. presented in that paper have shown that in cases of 10 to 20 seconds delay, a third of the customers began dialling before hearing the tone (resulting in a partial dial or false start). Another 10 percent, who did wait those 10 to 20 seconds, disconnected without dialling after a tone was provided. The control system will spend a considerable amount of its precious capacity on these unsuccessful calls. It is shown that the probability of successfully completing a call decreases rapidly as dial tone delay increases beyond a few tenth of a second.

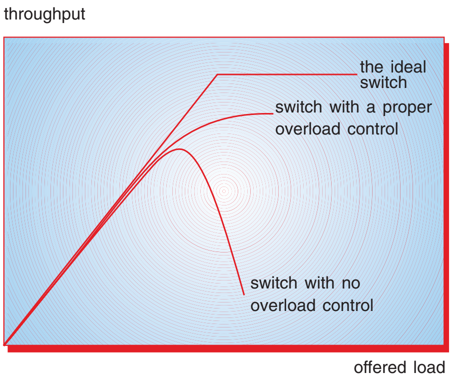


Figure 1

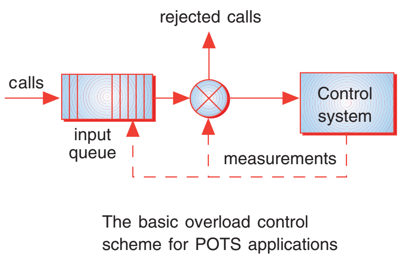


Figure 2.

Overload control strategies operating with mechanisms which throttle the input process are often used in communication systems, especially in today's SPC telephone switching systems. In these systems the complete mechanism is normally operating locally, i.e. entirely in the system that is to be protected. (See Figure 2.) Arrivals, throttled due to an overload situation, are normally stored in an input queue for later transportation to the processor queue(s). Measurements on the processor load, on the processor queue length or on the actual call arrival process are used to determine in what pace calls are fetched from the input queue to the processor queue. The input queue, which of course is limited, may however be congested and thus calls are lost. This rejection of calls is in most cases the way to handle these situations. Also under overload, accepted calls should experience a tolerable delay. The queuing discipline of the input queue could be LCFS, which under overload leads to those calls that pass the input queue experiencing a rather short delay while the others are just pushed out.

The importance of efficient and robust mechanisms for protecting control systems in public switches from overload will increase when new technologies, systems and services are introduced. These are switches for high speed integrated communications in the transport network. The introduction of B-ISDN will give rise to a number of new services imposing quite different loads and load profiles on the control systems. These are also switches in the packet switched signalling network capable of supporting real-time transaction oriented exchange of information between the different nodes dedicated to different tasks – Signalling System No. 7 is used for this purpose. They are also the SCPs (Service Control Points), i.e. the nodes that will host service logic and databases for today's services like freephone and televoting, but more importantly for a large number of advanced, complex and frequently used services of tomorrow. Finally, they are the HLRs (Home Location Register), which will not only host subscriber information and subscriber "profiles", but also a number of related services.

There is a clear trend that the control systems of these different types of switches and nodes will in many cases be implemented in terms of a distributed architecture with several processing elements. Such architectures have several advantages, first in terms of a modular software implementation, but also in terms of high call-handling capacity and linear growth facilities. Whether the control system in such an environment is based on a number of loosely coupled fully distributed processing elements or on elements individually dedicated to different tasks, the design of a traffic overload control strategy presents a number of challenging problems not encountered in a traditional centralised switch architecture.

Many investigations of overload strategies for SPC-systems have appeared during the last decade but still several basic questions remain to be answered. Some basic objectives of overload control are: to prevent switching capacity from decreasing as a result of increasing overload, i.e. the system should continue to show good throughput characteristics also under periods of overload; to assure that waiting times of accepted calls are within tolerable limits and not much larger than those under periods of normal load; to assure that there is no breakdown due to overload, i.e. processor queues being congested. In addition, it is generally required that some processor capacity must be saved for jobs other than those associated with pure switching. This is also important under periods of overload, which if sustained for a long period of time, could ultimately endanger system sanity, because system management, maintenance and audit functions would be entirely throttled.

The important question is how these goals should be met and fulfilled. What data should be collected to form a suitable basis for control decisions (processor load, queue length, call arrivals, call categories, etc.)? How should control algorithms using these data be designed? Should different services be controlled individually or by a common algorithm? In a distributed environment, should the overload control mechanism be implemented in each processor or should this mechanism be centralised?

Other basic requirements on mechanisms for overload control are: The amount of processor capacity needed to reject calls must be small. Thereby as much as possible of the processor capacity can be used for useful jobs. Call delays should not increase due to the overload control mechanism under periods of normal load. The overload control mechanism should be straightforward, robust and predictable and system performance not too sensitive to different parameter settings.

There are other major internal functions in the control system that may interact with the overload control mechanism. The monitor that handles the variety of tasks and sees to it that time critical tasks are given a higher priority than others is perhaps the most striking example. It is important to grasp the impact and limitations on overload control mechanisms from such a monitor. This in turn emphasises that the design of overload control mechanisms should be done in parallel with the design of the control system itself and not as late "patches". Good real-time performance can best be achieved by careful consideration of real-time issues early in the design process.

2 The new challenge

Most existing overload control schemes in public switches derive from pure telephone switch applications. These schemes are normally not well suited to cope with data traffic nor with a mix of ISDN traffic including packet switched signalling. In many of today's ISDN switches overload control schemes handle D-channel traffic and signalling only in a rudimentary way. The great variety of traffic load mixes to be handled by the switches may talk in favour of individual solutions for different exchanges based on individually controlled services. For obvious reasons, however, such designs are far from attractive: very detailed data collection is required; many control parameters must be defined and adjusted to a varying traffic mix. Indeed, it seems that great efforts should be spent searching for control principles that are not only efficient but also as robust and simple as possible.

Future control systems will in many cases be implemented in terms of a distributed architecture with a number of processing elements. In general, each processor has its own overload detection and control mechanism, since different overload situations may affect different processors. Then it is important to consider a global efficiency of the control system. In the case of a fully distributed architecture with loosely coupled processors, where each processor may be able to handle any task or service request to the system, the load sharing procedure has a strong impact on the overload control mechanism. In earlier papers [1] we have shown that certain load sharing principles, like Shortest Queue First, may under some circumstances lead to an undesired behaviour of the control systems. Heavy oscillations might appear in the job queues at each processor, which lead to a large increase in the mean waiting times. Again, good real-time performance can best be achieved by careful consideration of real-time issues early in the design process.

In Intelligent Networks the definitions of services reside in a few nodes in the network called Service Control Points (SCP). (See Figure 3.) It is very important to protect the SCPs from being overloaded. A network where call control, service control and user data reside in not only different nodes requires the support of a powerful signalling system capable of supporting real-time transaction oriented exchange of information between the nodes dedicated to different tasks. SS No. 7 is used for this purpose. Due to the construction of the involved communication protocols, the throttling of new sessions must be done at their origin, i.e. not at the SCPs, but in the SSPs (Signal Service Points). To manage the transfer of information between different nodes in an IN, SS No. 7 uses a set of protocols and functions on OSI-level 7 called Transaction Capabilities Application Part (TCAP). TCAP makes it possible to set up dialogues between nodes and to interchange information. Quite a lot of processing time must be spent on unwrapping the SS No. 7 protocol and TCAP before the contents of a message can be read. For example, in the freephone service, about half of the total processing time in the SCP is used to unwrap the protocols. Under overloads, the SCP will spend its real time capacity on

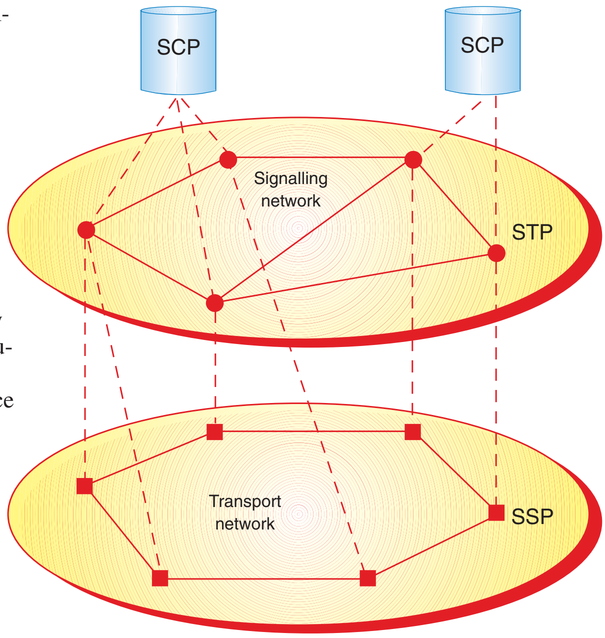


Figure 3

just unwrapping the protocol of arriving messages and discarding queries. Thus the SCP cannot protect itself from being overloaded. A node can only protect itself if the processing time needed to decide if a query should be accepted or not is small. Traditional approaches to load control will not be sufficient for the SCP nodes. Some of these difficulties can be solved by placing throttles in the nodes that initiate the work to be done by the SCP nodes instead of placing the throttles in the SCP nodes themselves. The flow control and rerouting in SS No. 7 are believed to have less influence on overload control implementation since these work on a shorter time scale than the overload control. Within the coming three to five years we will see STPs (Signal Transfer Points, i.e. switches in the signalling network) switching hundreds of thousands packets per second corresponding to some thousands of new sessions per second. If not properly protected from overload, the precious real-time capacity of the control system in the STPs will fade away.

A part of the SPC capacity might be reserved to one or several service providers or customers for a certain period of time. In other cases this reservation is permanent. Under periods of overload it is important to portion out the SCP capacity according to the reservations made. The service provider that can give

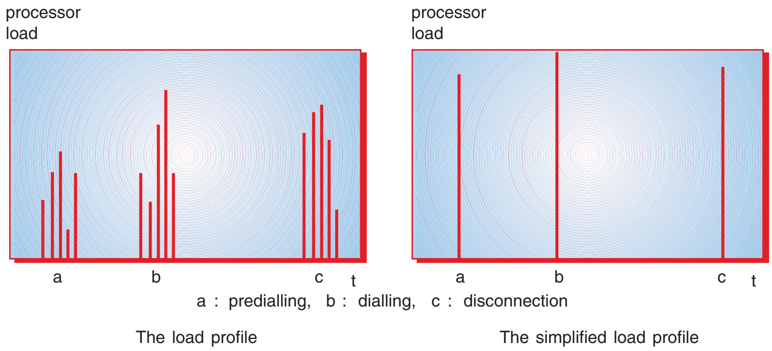


Figure 4

its customers good service under periods of overload might gain a lot. This further complicates the overload control scheme.

The original IN concept, separation of service control and switch control, was motivated by a need for rapid and independent evolution of services and technology and a need for advanced addressing facilities. Today, we see the IN concept as an important platform for networks of tomorrow. IN is the key to UPT, Universal Personal Telecommunication (not T for Telephony!), which will provide true personal mobility, not only for voice services but for B-ISDN. The HLRs, today located in the Mobile Switching Centers but tomorrow within the UPT (Universal Personal Telecommunications) switches, will be exposed to heavy traffic both in terms of queries from the "own" network and from other networks and signalling for the networks to operate properly together. B-ISDN will most likely offer a wide range of multimedia/multiconnection services needing complex call handling functions. UPT will definitely call for high-speed services. Walkstations want to reach the fixed ATM-network via high capacity radio channels. In this perspective IN will be one of the key elements in the B-ISDN/ATM network. So far, the research and standardisation on IN and B-ISDN have progressed more or less independently. IN architectures proposed by standardisation bodies (CCITT/ ETSI) are not taking into account B-ISDN service requirements [2], standardisation of a control architecture and signalling for B-ISDN have not taken advantage of the results obtained from the work on IN. An evaluation of the B-ISDN service requirements leads to a control architecture which enables calls to be established and controlled independently of associated connections. This implies separation of service handling and connection handling functionality. A strong relationship exists between the basic IN concept and the concept of service and connection control separation in the B-ISDN control architecture. However, the IN architectures proposed by standardisation bodies will need some modifications to incorporate B-ISDN requirements.

3 Overload control policies and principles

First it is important to point out that the internal overload control methods used in an exchange depend on the particular technical arrangement of the switching system. Thus, the very implementations of overload control schemes in the switches are questions for the vendors. However, users, operators, suboperators and also vendors have a number of requirements on the system as a whole, requirements that may in many cases only be fulfilled or expressed by setting requirements or guidelines to provide service during overload conditions.

In general, one may state that the overload control schemes are there just to ensure a high throughput. In order for the overload control scheme to ensure high throughput waiting times must be kept short, otherwise customers (humans or machines) will tend to behave unfavourably, leading to a degradation of throughput. It has been shown in many investigations, often mainly based on pure telephone applications, that customer impatience or persistence when blocked or when waiting times are too large, may lead to a totally congested control system in that repeated call attempts further overload the system. So the overload control scheme must be stable also under a number of different and often unpredictable customer behaviours. There is a number of other things that the overload control mechanism has to see to, like sparing some processing time for maintenance and communication with other stations. The man-machine communications essential for priority operational tasks should always be preserved. In particular, network management terminals and functions associated with interfaces to network management support systems should be afforded high priority, since network management actions may play an important role in reducing exchange overloads. If this capacity is not given to these often not that demanding real-time functions, throughput will decrease on a perhaps longer time scale. Nevertheless, many activities non-essential to the handling of offered traffic can be deferred without endangering the system. Accepted calls or data sessions should be completed and be given a good service. Priorities should be assigned to specific exchange measurements, such that low priority measurements cease at a predetermined level of congestion. Higher priority measurements may be ceased at a higher level of congestion, or may be run continuously, depending on their importance to the call handling functions. It is often said that the overload control scheme shall give preference to the processing of terminating calls. This is understandable. It shall also give preference to calls already being processed, before accepting new calls. Here, one has to be careful. As shown in several papers lately, serious synchronisation effects might under certain circumstances occur when distributing priorities of this kind, [3] and [4]. These synchronisation effects will lead to heavy oscillations in the buffers that in turn will run the system into a very serious state of congestion. It is also important to be able to give preference to priority class lines, calls to priority destinations based on digit analysis and incoming calls with priority indications in, for example, the Initial Address Message of a call using CCITT Signalling System No. 7, if an essential service protection capability has been invoked. Finally, the overload control mechanism in itself may only to a minor extent increase the load on the control system. Especially, no resources should be spent on call-handling tasks that do not lead to fulfilled calls.

It is important to keep the overload control mechanism as general as possible. We know that different services put different loads on the control system, but also different load profiles. For instance, in the SSPs, whether STM or ATM based, the load profile (time dependent demands on control capacity during the session) of a data connection might differ significantly from that of a telephone call. In the same station, the signalling traffic, which might vary very much depending on whether this station has a number of local exchanges connected or not, will definitely put a load on the control system very much different from the connections in the transport network. However, we require an overload control mechanism that does not rely on a large number of parameters that has to be set for each individual station as a result of the traffic mix, which will change anyhow. New services will be introduced constantly and that should not lead to parameters to be changed in the overload control schemes.

We want to separate functions called overload control from functions called overload protection. The latter functions are there to ensure system survival under overload and these functions may take actions to prevent that the system breaks down. Under these circumstances one may have to accept that for shorter periods high throughput can not be accomplished. Actions we find here are normally quite drastic: take down calls or sessions under the call or session establishment phase; stop scanning certain devices, etc.

Requirements for the new technologies, systems and services do not differ significantly from those set up for today's systems for voice and data applications. However, the requirements will be much harder to meet. It is not only the compressed time scale in that we have a number of high speed systems, but also the increased system complexity that will make it harder to design proper overload control schemes.

For many applications, not least those concerning the STPs in the signalling network and the SCPs in IN, calls or sessions that are to be rejected due to overload should be so at the point of origin. Take for instance the case when an SSP is sending a request to an SCP. Here the TCAP protocol is normally used. Usually several TCAP dialogues have to be set up between the SSP and the SCP to complete a session. If an SCP is protected from congestion only by rejection of dialogues, it would experience a serious throughput degradation during heavy overload due to the fact that the SCP will spend much of its processing capacity on unwrapping TCAP frames and then reject the corresponding session. If the process of rejection takes place already in the SSP much would be gained.

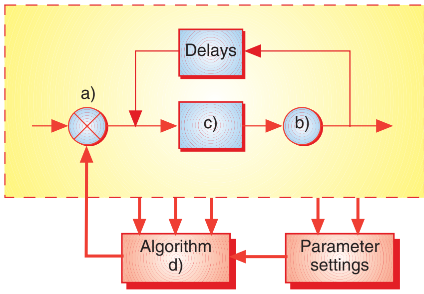
Many of the schemes used are based on either a Call Gapping mechanism, a Percentage Blocking mechanism or a Window Flow mechanism [5]. Each of these can be implemented locally (in the example above, in the SCP) or at the point of origin (in the SSP). Each of these may use one of the three metrics: load, queue length or number of arrivals, as the control parameter. However, when implemented in the SSP, these metrics may be forwarded from the overloaded system to that where the algorithm is implemented. If not, the algorithm can throttle the flow of calls based on information that calls are lost in the SCP.

4 Models

To understand the effects of different overload control mechanisms on switch as well as network performance, probabilistic models are of importance. These models are of great value when evaluating and comparing different overload control schemes. They are also used for dimensioning, i.e. for the setting of parameters in the schemes.

Several important contributions to the problem of overload control have been reported in the literature. In fact ITCs (International Teletraffic Congress) have been one of the more important fora for new ideas and solutions to the topic. The main part of the papers on overload control at the ITC congresses has been devoted to single processor systems and the analytical approaches have mostly been focused on stationary behaviour.

Transient solutions to models describing overload control are of vital importance, but not so frequently reported. This is natural since analytical solutions to such models are hardly obtainable. Nevertheless, it is of the utmost importance to deal with transient models, which naturally have to be solved numerically or be based on simulations. Lately, papers dealing with fluid approximation models have appeared. This technique has proven efficient in modelling a nasty oscillating behaviour that may occur in SPC-switches as a result of certain load batching effects, but could also be used for modelling the transient SPC-behaviour in general. Various overload control algorithms may perform equally well in a stationary system; still, their transient behaviour after, say a sudden increase of the arrival rate, may be quite different.



The feedback queue
a) : overload control throttle
b) : the processor
c) : the processor queue
d) : algorithm that opens/closes the throttle

Figure 5

The feedback queue model with delays in the different feedback loops has proved very useful in modelling the control system of a switch. (See Figure 5.) During a call or a session, the control system must handle a number of tasks. A number of tasks has to be processed during call initiating, another number of tasks during call establishment and finally some tasks during call disconnection. This is normally so whether it is a data session, a telephone call, whether it is STM or ATM, whether it is circuit switching or packet switching. Let us illustrate this by a simple example. A call enters the system, is put into the queue and receives its first service corresponding to pre-dialling work done by control systems. Then the call proceeds to a delay which models the time it takes to dial a number. After that the call returns to the queue and is given a second service corresponding to the work done at call set-up. The call proceeds to another delay that represents the conversation time and finally it returns to the queuing system where it is given a last service corresponding to call teardown before it leaves the system. This model, rough as it may seem, has been used successfully to study the behaviour of control systems and signalling systems, see e.g. [7] and [8]. Service times for the different executions will of course have different means and distributions. We think that also in the future similar models will be important when overload control mechanisms are studied. However, also new modelling techniques like e.g. neural networks will be needed.

As mentioned before, it is important to study non-stationary behaviour of overload control mechanisms. It is usually more difficult to obtain non-stationary solutions, but several powerful methods can be used.

Discrete event simulation is a standard technique for studying the steady-state behaviour of queuing models, but it can also be used to study transients. Suppose that *A(t)* is the value of some quantity at time *t* (e.g. the length of some queue or the load of some server) and that we want to study *A(t)* when $t \in P$ where *P* is a finite set of time points. The simulation program is run several times with different random number generator seeds. Each time the program is run we sample *A(t)* at the points in *P*. To get a high accuracy we need many samples of *A(t)* at each point in *P*, which means that we have to run the program many times. When we use simulation to obtain steady-state results we often have to make one long simulation run, when the transient behaviour is studied we instead make many short runs of the simulation program.

In some cases it is possible to obtain *explicit analytical expressions* for *A(t)*. A well-known example is $P_k(t)$ = probability of *k* customers at time *t* for an M/M/1 queuing system. In this case $P_k(t)$ can be expressed using infinite sums of Bessel functions. However, explicit analytical solutions showing the transient behaviour of queuing models are very difficult to obtain and thus almost never used in practice.

The derivation of $P_k(t)$ for the M/M/1 queuing system takes its starting point in the well-known Chapman-Kolmogorov differential-difference equations. It is possible to use e.g. Runge-Kuttas method to get numerical values of $P_k(t)$ without explicitly solving the Chapman-Kolmogorov equations. In many cases it is possible to write down differential-difference equations similar to those used for describing M/M/1 and then solve these equations numerically. We have named such methods *Markov approximation methods*. We often have to simplify the state space considerably to be able to handle the equations numerically.

If we only keep track of the mean values of all stochastic quantities we get a very simple state space. Often simple difference equations that approximately describe the time-dependent behaviour of the system can be obtained. We call such methods *flow approximations*. Such methods are rather crude but they can nevertheless give important insights into the transient behaviour of queuing models.

Simulation has many advantages. It is in principle easy to write and to modify simulation programs. We can use any statistical distributions, but execution times can get long if we want accurate results. The results are numerical, i.e. it is hard to see the impact of different parameters on system performance. Explicit analytical solutions showing the transient behaviour of queuing models are very difficult to obtain and are thus never used in practice. The numerical methods described above can give somewhat shorter execution times than simulation programs and in many cases standard numerical methods can be used. However, the state space usually has to be much simplified and it is often difficult to modify the model to test a new overload control mechanism. Fluid flow methods can only be used to study transients but has proven very useful not at least in modelling overload control schemes.

5 Conclusions

Though we foresee SPC-systems later during this decade with very powerful control systems, the need for efficient overload control mechanisms will increase. In this paper, we have briefly described the environment in which these switches will work. Though system complexity will increase in general, it is important to keep the overload control mechanisms predictable, straightforward and robust. We have stressed that the design of overload control mechanisms must be done in conjunction with the design of the control system.

We would like to point out the importance of studying time dependent behaviour of modern SPC-systems in general and especially their overload control mechanisms. This, indeed, seems to be a very urgent task. We have so far only reached a point of basic understanding of these mechanisms. It is also important to have the performance, derived from various analytic and simulation based models, supported by measurements from systems in use. Traditional stochastic models do suffer from limitations. We need to develop efficient numerical solution methods for many of today's models as well as to develop new models based on perhaps non-traditional approaches.

Acknowledgement

Many thanks to Dr. Christian Nyberg for valuable discussions and suggestions that have led to many improvements throughout this paper. The two of us have worked together for a number of years on the topic. Our research work is supported by Ericsson Telecom, Ellemtel Telecommunication Systems Laboratories and The National Board for Industrial and Technical Development (grant no. 9302820).

References

1 Nyberg, C, Wallström, B, Körner, U. Dynamical effects in control systems. I: *GLOBECOM '92,* Orlando, 1992.

2 Løvnes, K et al. IN control of a B-ISDN trial network. *Telektronikk,* 88(2), 52–66, 1992.

3 Erramilli, A, Forys, L J. Traffic synchronization effect in teletraffic systems. I: *The 13th international teletraffic congress, ITC 13,* Copenhagen, 1991.

4 Körner, U. Overload control of SPC systems. I: *The 13th international teletraffic congress, ITC 13,* Copenhagen, 1991.

5 Kihl, M, Nyberg, C, Körner, U. *Methods for protecting an SCP in Intelligent Networks from overload.* To be published.

6 Borcheing, J W et al. Coping with overload. *Bell Laboratories Record,* July/August 1981.

7 Skoog, R A. Study of clustered arrival processes and signaling link delays. I: *The 13th international teletraffic congress, ITC 13,* Copenhagen, 1991.

8 Erramilli, A, Forys, L J. Oscillations and chaos in a flow model of a switching system. *IEEE journal on selected areas in communication,* 9, 171–178, 1991.

Capacity of an Alcatel 1000 S12 exchange emphasised on the ISDN remote subscriber unit

BY JOHN YTTERHAUG, GUNNAR NOSSUM AND ROLV-ERIK SPILLING

1 General

This article describes the dimensioning of the main equipment of an Alcatel 1000 S12 exchange and the corresponding traffic restrictions where this is appropriate. It is divided into two main parts: The first part describes the Alcatel 1000 S12 exchange, and the second part describes the remote concentrator called ISDN Remote Subscriber Unit (IRSU). The IRSU is developed and maintained in Norway. The Alcatel 1000 S12 is a common Alcatel product with main development in Belgium. Each country may have different requirements for the equipment, reliability and grade of service. This article is based on the dimensioning rules for the Norwegian market.

Most of the Alcatel 1000 S12 exchanges in Norway have old software. The existing hardware in an exchange can with some adaptations be used with the new software package including ISDN, Business Communication, etc. Full functionality is therefore possible for all exchanges. All types of modules needed for this can be introduced, either in existing racks or by introduction of new rack types. The description here only concerns the latest versions of the modules and rack types.

2 Dimensioning of common exchange equipment

This part describes the main equipment in an Alcatel 1000 S12 exchange and the capacities and dimensioning of the equipment where appropriate.

2.1 Digital Switching Network (DSN)

An exchange is built up around a Digital Switching Network (DSN). The DSN transports circuit switched, packet switched and signalling traffic. All the modules in the system are connected to the DSN as shown in Figure 1. The modules are connected via Terminal Sub Units (TSU). Two access switches are equipped per TSU, and they connect the modules to the switching planes in the DSN.

The following modules in Figure 1 are not described later in the text:

TTM: Trunk Testing Module
CTM: Clock and Tone Module.

The number of switching planes is 3 or 4 depending on the traffic. All switching planes are identically equipped. The number of switching units in each plane is determined by the number of TSUs.

2.2 Terminal Sub Units (TSU)

A TSU consists of one group of terminal modules. One TSU is connected to one Access Switch (AS) pair. Figure 2 shows how two TSUs are connected to two Access Switches. The figure only shows the processors (TCE is Terminal Control Element) of the terminal modules.

Each AS is connected to all four switching planes in the DSN.

The number of TSUs is calculated based on the total traffic demand and the total amount of terminal equipment.

The maximum allowed mean traffic per TSU (determined by the GOS requirement for end-to-end blocking probability in the switching network) for 4 planes is 152 Erlang. This ensures that the signalling messages between the modules have a high probability to succeed. If a free path is not found, several reattempts are done, the number of reattempts depending on the importance of the message.

The number of subscriber modules and other traffic dependent modules per TSU is calculated based on the TSU traffic constraints given above.

2.3 Subscriber modules

Two types of subscriber modules exist, one for analogue and one for digital (ISDN) subscribers. The maximum traffic per subscriber module is 35.2 Erlang. The subscriber module is connected to the DSN via two separate PCM systems. With a GOS of 0.2 % the traffic capacity is 17.6 Erlang per connection. If the required traffic per line gives a higher total traffic per module than 35.2 Erlang, the subscriber modules will be underequipped.

Modules for analogue and digital subscribers may be combined in the same TSU. 2 modules with the same type of subscribers work as cross-over pairs, which means that if one module fails another will take over the function. The capacity, however, will be reduced to the half.

No hard restrictions exist on the penetration figures for subscriber facilities. Extending the requirements for capacity and availability is always possible by proper dimensioning of the required resources. It is possible for all subscrib-

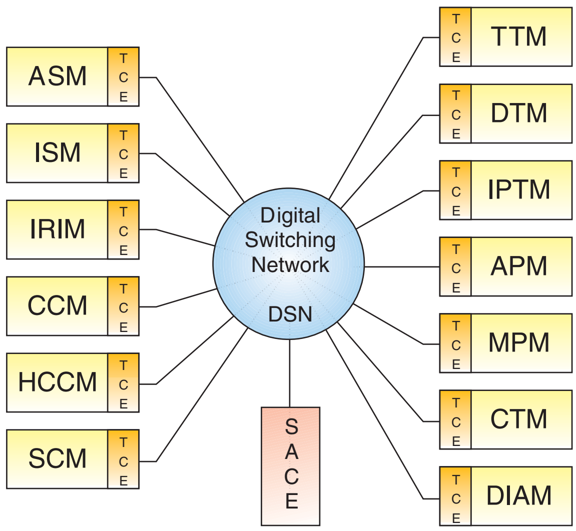


Figure 1

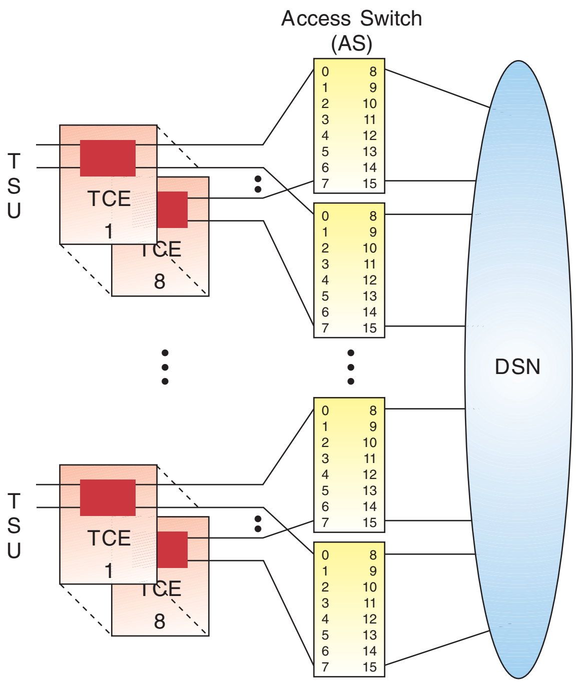


Figure 2

ers to use the same facility at the same time provided the dimensioning has been done accordingly. One subscriber can have a combination of facilities and can use his facilities in the same call as far as they are not incompatible. The influence of combinations on the dimensioning can be neglected.

2.3.1 Analogue Subscriber Module (ASM)

The module terminates 128 analogue lines, with a maximum traffic of 0.275 Erlang per line in a fully equipped ASM. An ASM consists of two common PBAs (processor and ring PBA) and from 1 to 8 subscriber PBAs with 16 analogue lines per PBA.

2.3.2 ISDN Subscriber Module (ISM)

The module terminates 64 Basic Access (BA) lines, with a maximum traffic of 0.550 Erlang per line in a fully equipped ISM. An ISM consists of one common PBA (processor) and from 1 to 8 subscriber PBAs with 8 ISDN lines per PBA.

2.4 High Performance Common Channel Signalling Module (HCCM)

The exchange has several possibilities for terminating No. 7 links. The most common is the use of the HCCM module which controls up to 8 external No. 7 links. This module consists of one processor PBA and one line PBA per No. 7 link.

The signalling terminal is capable of receiving and sending N7 messages with only one flag between the messages. The two-way throughput capacity under normal load conditions is 500 messages per second per link. The transmission capacity is 64 kbit/s in each direction.

With a message length of 20 bytes this will give a load of 0.625 Erlang on the No. 7 link.

With normal holding times and utilisation of the 2 Mbit/s one No. 7 link can signal for several hundred 2 Mbit/s with today's type of traffic.

2.5 Trunk modules

Different types of trunk modules are provided and they are described in the following sections.

2.5.1 Digital Echo Cancelling Module (DEC)

One module with integrated echo cancellation equipment terminates one first order PCM link with different versions for No. 5 signalling and No. 7 signalling. One module consists of two PBAs (one processor and one echo cancelling PBA).

2.5.2 ISDN Packet Trunk Module (IPTM)

This module has several applications. The hardware is the same, but the software varies. Each of the IPTM modules consists of two PBAs (one processor and one trunk PBA).

- Primary Rate Access (PRA)

  This module can be used for termination of an ISDN PABX connected with a PRA connection.

- Common Channel Signalling (CCS)

  This module has a lower capacity per link than the HCCM module, and is more cost efficient if only a few No. 7 links are required. The module terminates up to 4 No. 7 channels in a PCM link. The rest of the channels can be used as traffic channels. The capacity of the signalling terminal under normal load conditions is 400 messages per second per module.

- Frame Handler (FH)

  The frame handler part comprises concentration and multiplexing of packet switched traffic. This module terminates up to 4 packet channels in a PCM link. The rest of the channels can be used as traffic channels. The two-way throughput capacities under normal load conditions is 300 frames per second per module.

- X25

  This module terminates up to 4 packet channels in a PCM link.

- Packet Handler Interface (PHI)

  This module is used as a gateway function for the Packet Handler in DATAPAK.

2.5.3 Digital Trunk Module (DTM)

One module terminates one first order PCM link with channel associated signalling or transparent No. 7 signalling. The module consists of only one PBA.

2.5.4 ISDN Remote Interface Module (IRIM)

One IRIM/S, respectively one IRIM/T, is used for termination of 1, respectively 2, links to a singledrop (SD) IRSU or a multidrop (MD) of IRSUs. 2 IRIMs work as a cross-over pair. The module consists of two PBAs (one processor and one trunk PBA, DTRF for double trunk and DTRH for single trunk).

2.6 Dynamic Integrated Announcement Module (DIAM)

The new announcement module DIAM will provide the following types of announcements:

- Fixed announcements (e.g. Call Forwarding)
- Combined fixed and variable announcements (e.g. Changed Number Interception)
- Variable announcements (e.g. Personalised Changed Number).

Two versions exist:

- DIAM-basic, one PBA version. Requires 4 Mbyte storage capacity on disk. Speech storage capacity is 8 minutes
- DIAM-extended, two PBAs version. Requires additional 20 Mbyte of storage capacity on disk (a total of 24 Mbyte for the whole module). Speech storage capacity is 50 minutes.

The number of speech channels are:

- 56 channels can be used for speech
- 4 channels are reserved for virtual path communication.

The maximum duration for one announcement is flexible and only limited by the speech store capacity. If more announcement capacity is needed per exchange, more modules may be equipped.

2.7 Service Circuit Module (SCM)

The modules are dimensioned in separate groups per signalling type. Within each group a cyclic selection mechanism with 4 trials is used to find an SCM with free service circuits. For reliability reasons only 3 trials is assumed for dimensioning purposes.

Groups containing R2 or R5 signalling units are dimensioned according to the

GOS requirement of 0.1 % waiting probability for seizing of a free circuit.

For exchanges without MFC signalling, the GOS requirement for DTMF/R2 signalling units will be 1.0 %.

The modules will be equipped in different modes:

- One MF-unit with 16 senders/receivers for R2/DTMF and one Simplified Conference Bridge (SCB).

  The SCB can handle 6 simultaneous conferences with maximum 5 participants per conference. The number of SCB units is calculated based on the originating traffic using the services Add on Conference and Meet-me Conference and a part of the originating traffic using the services Three Party Service, Call Hold and Call Transfer. The GOS requirement is 1.0 %.

- One MF-unit for R2/DTMF and one service unit for Continuity Check Transceivers (CCT). CCT equipment is used for No. 7 traffic on international routes. One CCT service unit contains 16 CCT devices.

- Two MF-units for R2/DTMF with a total of 32 senders/receivers.

- One MF-unit for R5 and one service unit for CCT (Continuity Check Transceivers).

Each of these SCM types will consist of only one PBA in addition to the processor. As a general rule, an extra module of any equipped type will be added for reliability reasons if the number of modules of that type is less than or equal to 4.

2.8 Maintenance and Peripheral Module (MPM)

This module contains functions for peripherals and loading and works together with the magnetic and optical disk. This module also contains functions for administration. Two modules are equipped per exchange.

2.9 System Auxiliary Control Elements (SACE)

A minimum number of SACEs are required per exchange. For new SW releases the installed processors are reused. If not explicitly given, the following traffic is assumed:

- Centrex lines: 5 % of all analogue subscriber lines and basic accesses

- Traffic per Centrex line: 0.2 Erlang

- PABX lines: 10 % of all analogue subscriber lines and basic accesses

- Traffic per PABX line: 0.4 Erlang.

In the following sections the SACEs are described.

2.9.1 Call Service SACE

This SACE contains the following functions:

- Prefix analysis and task element definition

- Local subscriber identification

- Charging analysis

- Call control

- Charge generation control

- Trunk request co-ordination

- Facility processing

- Alarm call processing.

The Call Service SACE works in load sharing mode for all functions. Therefore, the Call service SACE has to be split into load sharing groups. With service penetration figures used for Norway, a load sharing group has a capacity of 9600 subscribers.

For security reasons all data are replicated over at least two processors, which are working in load sharing for processing the data accesses. Each load sharing group is equipped with one extra processor if the number of processors calculated is less than 5.

2.9.2 PBX/CHRG SACE

This SACE contains the functions to manage PBX lines (analogue and digital), Centrex hunt groups and the charging function.

PBX/CHRG SACE contains the following functions:

- Automatic message accounting
- Meter counts collection
- Local tax layout
- Private access resource management
- BCG resource management
- Call control for PABXs
- Charge generation control for PABXs.

The PBX/CHRG works in active/standby.

2.9.3 CCSN7/DEF SACE

This SACE contains the defence, No. 7 and operation and maintenance functions. In addition, this SACE contains functions required for interfacing a taxation centre. One pair working in active/standby mode is equipped.

2.9.4 IN/OSI SACE

This SACE contains IN call control and OSI-stack modules working in load sharing mode. An extra processor is equipped if the required number of processors is less than 4.

2.9.5 IDC/TRA SACE

This SACE contains the following functions:

- Line intermediate data collection
- Trunk intermediate data collection
- Trunk request allocation
- Device interworking data collection.

The processor load is calculated for each function. 5 % of the processor capacity is reserved for storage of measurement counters on disk, assuming a storage interval of 15 minutes. One extra processor is always equipped.

2.10 Network Services Centres (NSC)

If the exchange has a collocated NSC, an Administration and Peripheral Module (APM) has to be added.

This module contains functions for administration and peripherals in addition to the specific NSC-function. These two functions are not split into separate processors. The APM works together with magnetic and optical disk. Two modules are equipped per NSC and are the same as the MPMs.

2.11 Taxation Centre

A Taxation Centre consists of a Transaction Processor (TP) and a Billing Processor (BP). The TP is an integrated part of the exchange. Each TP can handle 64 exchanges or 100,000 subscribers.

The TP is a duplicated processor system where the Administration and Peripheral Modules (APM) work in active/active mode.

3 ISDN Remote Subscriber Unit (IRSU)

The purpose of the IRSU is to provide a cheap subscriber connection to an Alcatel 1000 S12 exchange. As the name Remote Subscriber Unit indicates, the use is primarily for clusters of remote subscribers relative to the exchange. The advantage

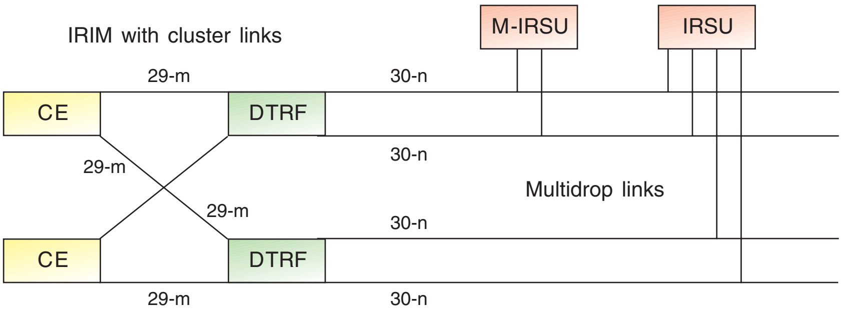


Figure 3

of introducing an IRSU is a considerable saving on cables if the alternative is to connect each of the remote subscribers to the exchange itself. If the alternative is to put a new exchange on the remote site, the solution of an IRSU is much cheaper and will save administration and maintenance costs since the administration of the IRSU is done from the host exchange. Only HW maintenance is necessary on the site of the IRSU.

An IRSU is connected to the exchange through an IRIM pair. Up to eight IRSUs may be connected in a series and is then called a multidrop (MD). One IRSU connected to an IRIM pair is called a single-drop (SD). An SD or an MD is connected to one IRIM pair.

An MD is used primarily for low traffic subscribers, typically a fully equipped MD can carry up to 0.083 Erlang per equivalent line. An analogue line and a digital line are one and two equivalent lines respectively. It may also be used for high traffic subscribers, but underequipping is then necessary.

It is important to note that an IRSU does not have a processor PBA that takes care of the call handling which is handled in the exchange in the same way as for directly connected subscribers.

3.1 Multidrop description

An MD consists of two half-systems, called the upper and lower side. A maximum of 8 IRSUs can be connected to the same MD system with a maximum of 1024 equivalent lines. A maximum of 4 subscriber groups of 128 equivalent lines may be assigned to each IRIM in the IRIM pair.

- The IRIM/S is equipped with a DTRH trunk PBA which terminates one PCM link (single trunk system).
- The IRIM/T is equipped with a DTRF trunk PBA which terminates two PCM links (double trunk system).

On the IRSU side the DTRH will usually be connected to an IRIM/S, while the DTRF will be connected to an IRIM/T.

An IRIM has only one processor. A maximum of 4 MD links can be assigned to one MD configuration (2 x IRIM/T).

The upper and lower side is cross-connected by cross-over links in the IRSUs as well as in the IRIM modules except for one trunk PBA configurations (such as mini IRSU and self restrained configurations). This assures that from each subscriber all trunks can be reached.

Available standard types of PBAs are used for connecting the subscribers. Between indicated maximum limits, any 16 analogue subscribers can be replaced by 8 ISDN subscribers.

3.2 IRSU variants

Each variant consist of one subrack containing one (JR01), two (JR02) or three shelves (JR03). A common rack (JR00) has space for six shelves grouped three and three.

The JR04 rack (mini IRSU) will occupy one shelf in the JR00 rack.

The transmission equipment for one IRSU is provided by a separate subrack (JR06). The shelf variants have the following subscriber capacities:

- JR01:
  256 ASL + 0 BA to 32 ASL + 112 BA
- JR02:
  512 ASL + 0 BA to 32 ASL + 240 BA
- JR03:
  976 ASL + 24 BA to 64 ASL + 480 BA
- JR04:
  96 ASL + 0 BA to 0 ASL + 48 BA.

JR05 is an outdoor cabinet which has space for a JR01 rack, transmission equipment, termination for optical fibre cables, power supply and batteries.

3.3 Dimensioning of trunks in a multidrop

Dimensioning of trunks is done based on a GOS requirement of 0.5 % for the circuit switched (CS) traffic between the IRSU and the group switch in the exchange. The GOS requirement is for the ‘worst case’ subscriber. In the special configurations described later, the difference of GOS for subscribers can be substantial.

The capacities given later are valid for configurations with one packet switched (PS) channel per MD link. It is possible to allocate one PS channel on one MD link and none on another, but the traffic capacities for such configurations are not described. The allocation of a PS channel replaces a CS channel and vice versa. Thus, as the number of allocated PS channels increases, the CS traffic capacity is reduced.

The two main dimensioning parts of the system is the MD links which each has 30 channels and the cluster side links (see Figure 3) of the IRIM which has 29 channels each for circuit and PS traffic. The number of CS channels is shown in Figure 3. In this figure, m and n is the number of PS channels on the cluster links and MD links, respectively. For a DTRH configuration in the IRIM, $n = 1$ will imply $m = 1$. For a DTRF configuration in the IRIM, $n = 1$ will imply $m = 2$.

The IRSU has the same schematic structure as the IRIM. Instead of the Control Elements (CE, i.e. the processor) there are clock and alarm PBAs. Each subscriber PBA is connected to one direct and one cross-over link between the clock and alarm PBA and the DTRF/DTRH.

To fully understand the traffic capacities in the following sections, it is important to note that the traffic from a subscriber group on the upper side has to go through the upper processor (control element) in the IRIM and similarly for the lower side subscribers.

The normal channel allocation algorithm is first to try the direct route, i.e. from a subscriber on the upper to the upper control element in the IRIM through the direct cluster link. Only if this direct route is blocked, the cross-over cluster link in the IRIM will be tried. The same procedure is followed for subscribers on the lower.

3.3.1 Normal configurations

Traffic limits for the normal trunk configurations, which are the two link and four link configurations, are as follows:

Two DTRH PBAs: 43.0 E
Two DTRF PBAs: 82.5 E

In the case of two DTRH PBAs the MD links are the bottleneck, i.e. Erlang's formula for 58 channels is used. In the case of two DTRF PBAs the direct and cross-over cluster link in the IRIM are the bottleneck, i.e. Erlang's formula for 56 channels multiplied by two is used. These limits are based on the use of the same trunk PBAs in the IRSUs in the MD, i.e. that all IRSUs have 2 DTRHs or 2 DTRFs, respectively. If several IRSUs with DTRHs are connected to a DTRF in the IRIM, the DTRH PBAs should be distributed equally on the two MD links for capacity reasons.

Furthermore, it is assumed that each IRSU has subscribers on both the upper and lower side of the system, i.e. at least two subscriber groups per IRSU. In an MD configuration, the total traffic from all IRSUs has to be within the above given limits.

3.3.2 Special configurations

If the traffic from the IRSUs is much smaller than 43.0 Erlang or a little more than 43.0 Erlang, some special configurations may be considered. These configurations are normally not recommended due to reliability reasons or traffic reasons (because of an asymmetric configuration).

3.3.2.1 One trunk PBA configurations

The economical one trunk PBA configuration can be used if the traffic is low and the reliability aspect is not considered that important.

Figure 4 shows a one trunk PBA configuration with DTRF PBA in the IRIM and the IRSUs. The IRSUs have subscribers on both upper and lower side.

Traffic limits for the one PBA configurations are as follows:

One DTRH PBA: 18.2 E
One DTRF PBA: 34.8 E

In the case of DTRH PBA the MD link is the bottleneck, i.e. Erlang's formula for 29 channels is used. Very unbalanced traffic is well tolerated. For example there can be 17.0 Erlang from subscribers on the upper side and 1.2 Erlang from subscribers on the lower side and vice versa.

In the case of DTRF PBA the bottleneck is the direct and cross-over cluster link in the IRIM, i.e. Erlang's formula for 28 channels multiplied by two is used. It must here be assumed that the traffic on the upper and lower is well balanced.

If the trunk PBA in the IRIM or the trunk PBA in the IRSU fails, the connection between the IRIM and the IRSU will be disrupted, whereas for a normal configuration the traffic will use the alternative cross-over link when the direct link is disrupted.

3.3.3 Asymmetric configuration

This configuration has one DTRF PBA and one DTRH PBA in the IRIM and in the IRSU as well, i.e. three MD links. Figure 5 shows an asymmetric configuration.

The asymmetric configuration may seem a good solution for traffic needs between the traffic capacity for a configuration with two DTRHs and two DTRFs in the IRIM. But it should only be used with the alternative channel allocation algorithm which chooses the MD link pair with most free channels.

Without this alternative channel allocation algorithm the capacity can be improved by placing more subscribers on the side (upper or lower) where the DTRH is placed, which at first may seem surprising. When the traffic on the DTRH side is twice the traffic on the DTRF side the traffic capacity is 61 Erlang. In this case the high traffic from the DTRH will be blocked on the cluster link, but different from the opposite traffic situation the overflow traffic together with the direct traffic will now have two MD links available on the DTRF side. There will be very little overflow traffic from the DTRF side to the DTRH side. The point is to distribute the traffic so that the cluster links carry equal traffic as much as possible.

If the alternative channel allocation algorithm is used the performance of the balanced upper and lower traffic case is improved. This algorithm has the effect of distributing the traffic more evenly and thus reducing the more bursty overflow traffic. With balanced traffic on the upper and lower side the traffic capacity is 66.3 Erlang and is dimensioned according to Erlangs formulae with 87 channels minus PS channels. So, if this alternative channel allocation algorithm is used the asymmetric configuration may be considered.

3.4 Mini IRSU

The mini IRSU is an inexpensive, simple IRSU. The mini IRSU is special in two respects. It contains only one subscriber group and it has only one trunk PBA (DTRH or DTRF) which means it has no cross-over link. This means that a mini IRSU subscriber is a 'worst case' subscriber when the mini IRSU is in an MD with other non mini IRSUs. To fulfil the GOS requirement for all subscribers in an MD it is necessary to decrease the total traffic if the mini IRSU traffic increases. The traffic capacity for the mini IRSU is 17.4 Erlang. The bottleneck is the cluster link, hence Erlang's formula with 28 channels is used. However, the bottleneck is quickly changed to the MD link with 29 channels when other non mini IRSUs are connected to the same MD and there are DTRHs in the IRIM. If there are DTRFs in the IRIM the bottleneck is again the cluster link.

Figure 6 shows a configuration with DTRH PBAs in the IRIM and the IRSUs

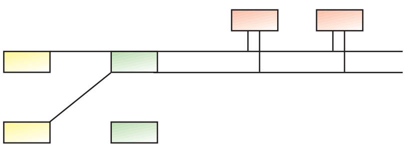


Figure 4

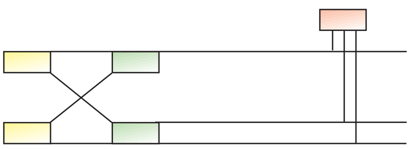


Figure 5

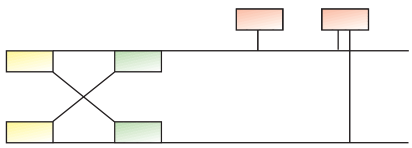


Figure 6

and one mini IRSU and one normal IRSU in the MD.

The mini IRSU is an economical way to introduce ISDN subscribers if there are IRSUs connected to the exchange already.

The rest of this section is based on an alternative channel allocation algorithm which chooses a channel on the MD link pair with most free channels. This has the effect of distributing the traffic more evenly on upper and lower MD links.

If one mini IRSU is placed on the upper and one is placed on the lower, the total capacity for the MD is 34.8 Erlang regardless of which trunk PBAs that are equipped and whether there are other than mini IRSUs or not in the MD. Thus, it is better to place all the mini IRSUs on one side (if there are fewer than five mini IRSUs). This is particularly important in the case of DTRFs in the IRIM. If this is done, the total traffic capacity for the MD depends on the traffic from the mini IRSUs and the more the traffic from the mini IRSUs approaches the limit of 17.4 Erlang the less traffic the MD as a whole can carry.

Since there is no bursty overflow traffic that mixes with normal traffic it is easy to find analytical models. In the following formulae this notation is used:

- *Am_u:* Traffic from mini IRSU which is connected to upper side
- *Au:* Traffic from upper side (excluding mini IRSU traffic)
- *Al:* Traffic from lower side
- *Lc:* Traffic limit on one cluster link
- *Lm:* Traffic limit on one MD link

If there are DTRHs in the IRIM the MD link on the side of the mini IRSU is the restricting part of the system for most distributions of *Au* and *Al.* Only if *Al* is far higher than *Au* (under the assumption that the mini IRSU is placed on the upper side) will the restricting part of the system be moved from the MD link to the upper direct cluster link. The following formula applies assuming *Au* and *Al* are approximately equal:

$$Am\_u + \frac{\frac{Au+Al}{2}}{\frac{Au+Al}{2} + Am\_u} * \frac{Au + Al}{2} \leq Lm$$

If *Au* = *Al* the formula can be reduced to:

$$Am\_u + \frac{Au^2}{(Au + Am\_u)} \leq Lm$$

If there is 10.0 Erlang coming from the mini IRSU then *Au* + *Al* must be less than 28.1 Erlang.

Simulations show that the limits given by this formula is conservative for all values of the traffic from the mini IRSU. This is mainly due to less burstiness than ordinary Poisson traffic for the traffic on the MD links from the ordinary IRSUs. A secondary effect is that high mini IRSU traffic will ‘push’ the other traffic over to the other MD link which will help the throughput of mini IRSU traffic.

For dimensioning purposes the interesting traffic interval for the mini IRSU is between 5 and 15 Erlang. As a uniform rule in this interval the total MD capacity could be set to two times the traffic capacity on one MD link, i.e. 36.4 Erlang even though a higher traffic can be carried.

If there are DTRFs in the IRIM the direct cluster link on the side of the mini IRSU is the restricting part of the system for all distributions of *Au* and *Al.*

The following formula is given for the case of two DTRFs in the IRIM:

$$Am\_u + \frac{\frac{Au+Al}{2}}{\frac{Au+Al}{2} + Am\_u} * \frac{Au}{2} \leq Lc$$

If *Au* = *Al* the formula can be reduced to:

$$Am\_u + \frac{Au^2}{(Au + Am\_u) * 2} \leq Lc$$

The second part of the sum is the traffic from the normal IRSUs offered to the direct upper cluster link of the IRIM.

So, for example for 10.0 Erlang from the mini IRSU, the total traffic for a two DTRF PBAs configured MD is 53.3 Erlang assuming that the traffic from the other IRSUs are evenly distributed on upper and lower.

Simulations show that the traffic limits derived from the formulae above is conservative for all values of *Am_u* and more so the higher mini IRSU traffic. The total traffic capacity is reduced as the mini IRSU traffic increases.

Thus, the difference between the DTRH case and the DTRF case is that the total traffic capacity is more or less constant up to a certain mini IRSU traffic for the DTRH case, whereas the total traffic capacity falls continuously as the mini IRSU traffic increases for the DTRF case.

3.5 Dimensioning of PS channels in a multidrop

As mentioned in the ‘Dimensioning of trunks in an MD’ section the total number of circuit and PS channels is constant.

The nature of PS and CS switched traffic is very different. The CS traffic is a so-called ‘lost calls cleared’ system whereas the PS traffic is a so-called ‘waiting call’ system. The latter system does not clear the call if there are no recourses available at the moment. Each packet of a PS call is treated according to a ‘waiting call’ system.

The different nature of the two types of strategy for packet and CS traffic can be exploited by the network operator. It could be tempting to minimise the number of PS channels.

However, the delays for the packets cannot be allowed to be too long because of buffer size and protocols. If a packet is delayed more than a certain amount of time the packet is retransmitted and when many packets are retransmitted the traffic increases which in turn gives even longer waiting times.

3.5.1 The PS system

Some characteristics for the PS system are listed below:

- Receive and send side in the IRIM and IRSU work in parallel.
- Each PS channel is used only in one direction, from the IRIM through IRSU 1 to the highest numbered IRSU and back to the IRIM.
- Packets sent from the IRIM is received by one of the IRSUs and sent on the PS channel through the loop to the IRIM where it is absorbed. Packets sent from an IRSU is received and absorbed by the IRIM.
- Each IRSU and the IRIM are only allowed to send one packet at a time before possibly being interrupted by another IRSU or the IRIM.
- Each IRSU and the IRIM have one queue per PS channel.

3.5.2 Analytical model

To describe the system correctly there should exist one queue in every drop which experiences arrivals of PS calls both from its own subscribers and the previous drops in the MD and from the IRIM (incoming traffic).

There are assumed *n* M/M/1 queuing systems (Markovian input process, Markovian service time distribution, one server) with first in first out (FIFO) queue discipline per PS channel where *n* is the number of PS channels. The *n* channels serve the whole MD.

The server time is assumed to be proportional to the packet length with a service speed of 64 kbit/s. Two traffic models are given for the PS traffic, the medium traffic and the high traffic model. The medium and high traffic model has 10 and 50 call attempts per hour (CAPH) per subscriber respectively. Each call has an average of 18 packets with an average length of 125 bytes. Since both outgoing and incoming PS calls are queued successively in all IRSUs and the IRSUs are considered as one common queue, all calls are considered outgoing.

The maximum number of ISDN subscribers in an MD is 512. The results in the following sections are based on this number of subscribers. It is important to note that the tables reflect the total PS traffic equivalent to 512 subscribers and that realistic PS traffic will be much lower due to fewer ISDN subscribers in an MD.

3.5.3 Results

In the following text and tables the following abbreviations occur:

- λ: Packet arrival rate per second for a MD
- μ: Service rate in packets per channel
- CH: Number of PS channels
- CHT: Call Holding Time
- CAPH: Call Attempts Per Hour
- DELq: Average delay in milliseconds (ms) for a delayed packet
- DELall: Average delay in ms for all packets
- Qbyte: Average number of queued bytes per channel
- Load: Load per channel
- 95%_all: Delay in ms for which 95 % of all packets are below
- 95%_del: Delay in ms for which 95 % of the delayed packets are below.

From Table 1 can be seen that for medium PS traffic one or two packet channels are sufficient. For high PS traffic three or four channels are sufficient, but one or two channels is impossible since the load is 2.0 and 1.0 respectively (a load higher than 1.0 gives an infinite queue).

The delays depend on the packet length given the same load. In Table 2 the bytes per packet and packets per call are interchanged so that there are many short packets instead of few long packets. Short packets reduce the delays and average queue lengths. Since an objective is to minimise the number of long delays to avoid retransmission, shorter packets may be used.

3.5.4 PS traffic in the signalling channel

The PS traffic intensity generated by ISDN subscribers may be so small that an allocation of one separate PS channel, and hence a reduction in CS traffic capacity, would be an unsatisfactory utilisation of the system. Thus, the signalling (S) channel may be used as a combined channel for PS traffic and S traffic. This solution utilises the spare capacity of the S channel for PS traffic.

The same queuing model as in the section ‘Analytical model’ has been used.

The abbreviation Wpacket is used for the weighted average packet length in bytes based on the distribution of CS and PS packets.

The ‘2 CH’ and ‘4 CH’ columns in the tables correspond to a DTRH and a DTRF configuration respectively.

CAPH and CHT is based on 0.08 Erlang per subscriber.

CSpackets/call is 88 packets on an average, i.e. 88 S packets per CS call. CS-bytes/packet is 33 bytes on an average. This assumes that all traffic carried in the MD originates and terminates here, which is a worst case. Approximately 2/3 signalling is for the originating side. 1/3 is for incoming traffic. Since only few calls use maximal signalling and the rest is balanced, the theoretical results should be slightly conservative.

In Table 3 only S traffic is considered. The results in Table 3 clearly show that even with high S traffic the load contribution due to S traffic will be modest (worst case load-DTRH = 24.8 %). The average waiting time for all packets will not exceed 1.35 ms, and 95 % of the

Table 1 PS Traffic

CAPH/SUB	10		50	
Bytes/packet	125		125	
Packets/call	18		18	
λ	25.6		128.0	
μ	64		64	
	1 CH	**2 CH**	**3 CH**	**4 CH**
λ	25.6	12.8	42.7	32.0
Load	0.40	0.20	0.67	0.50
DELq	26.0	19.5	46.9	31.3
DELall	10.4	3.9	31.3	15.6
Qbyte	33.3	6.3	166.7	62.5
95%_del	78	59	140	94
95 %_all	54	27	121	72

Table 2 PS Traffic

CAPH/SUB	10		50	
Bytes/packet	18		18	
Packets/call	125		125	
μ	444		444	
	1 CH	**2 CH**	**3 CH**	**4 CH**
λ	177.8	88.9	296.3	222.2
Load	0.40	0.20	0.67	0.50
DELq	3.8	2.8	6.8	4.5
DELall	1.5	0.6	4.5	2.2
Qbyte	4.8	0.9	24.0	9.0
95%_del	11	8	20	14
95%_all	8	4	18	10

Table 3 Signalling Traffic

CAPH/SUB	3.2		9.6	
cht	90		30	
Bytes/packet	33		33	
Packets/call	88		88	
λ	40.04		120.16	
μ	242.42		242.42	
	2 CH	**4 CH**	**2 CH**	**4 CH**
Load	0.082	0.041	0.248	0.124
DELq	4.4	4.3	5.4	4.7
DELall	0.37	0.18	1.35	0.50
Qbyte	0.25	0.06	2.69	0.58
95%_del	13	12	16	14

Table 4 Combined signalling and packet traffic

CSpackets/call	88			
PSpackets/call	18			
CSbytes/packet	33			
PSbytes/packet	125			
	2 CH	**4 CH**	**2 CH**	**4 CH**
CScaph/sub	3.2	3.2	9.6	9.6
PScaph/sub	10	50	10	50
Total caph/sub	13.2	53.2	19.6	59.6
Bytes/Wpacket	68.9	103.1	49.2	80.4
λ (Wpacket/s)	65.6	168.0	145.7	248.1
μ (Wpacket/s)	116.2	77.6	162.7	99.4
Load	0.28	0.54	0.45	0.62
DELq	12.0	28.1	11.1	26.7
DELall	3.4	15.2	5.0	16.7
Qbyte	7.7	65.8	17.9	83.3
95%_del	36	84	33	80

delayed packets will not experience a waiting time longer than 16 ms.

Combining S and PS traffic through the parameter Wpacket, higher waiting times can be expected. Table 4 shows the effect of mixing PS traffic and S traffic, see also Table 1 which has the same PS traffic. The highest 95%_del has the DTRF configuration with 50 CAPH for the PS traffic and 90 seconds CHT (3.2 CAPH) for CS traffic. But the highest DEL_all is seen for the DTRF configuration with the same PS traffic and 30 seconds CHT for CS traffic (higher load).

The loads given in the tables do not give serious delays, i.e. delays that cause a lot of retransmissions. The link access protocol has a threshold of 1000 ms before a retransmission occurs and thus retransmission should be a rare incident. The model is very simple so with more bursty traffic, delays and queues may be longer than indicated in the tables, but since the margins are high the risk of jamming down the system with retransmissions should be minimal.

4 Summary

The dimensioning of an Alcatel S12 exchange is done according to the description in chapter 2. The dimensioning is done in a very flexible way.

A traffic increase, whether it comes from (new) ASMs or DTMs can easily be handled by increasing the number of SACEs, SCMs and other equipment.

The IRSUs in an MD can be configured in many ways as described in chapter 3. For example can one large IRSU or up to 8 smaller IRSUs be connected to an IRIM pair. In addition, the number of links (2 Mbit/s) connected to the IRIM-pair can vary, but the recommended number of links are two or four. In addition, a mini IRSU may be connected, but care must be taken because of special traffic limitations.

The D channel used by digital subscribers connected to an MD is normally connected to allocated PS channels on the MD links. One PS channel per link may be assigned. If the PS traffic is low, there will be a possibility to use the signalling channel both for signalling and packet traffic which means that the circuit traffic capacity is increased.

References

For a thorough, but not up to date description of Alcatel 1000 S12 the following references are recommended:

1 ITT 1240 Digital Exchange, *Electrical communication,* 56(2/3), 1981.

2 System 12 Digital Exchange, *Electrical Communication,* 59(1/2), 1985.

For a more thorough view on the Digital Switching Network and general dimensioning the following reference is useful:

3 Haukeland, J. Hvordan dimensjonere en System 12 sentral. *9th Nordic teletraffic seminar (NTS 9).* Kjeller 1990.

Structure and principles of Telenor's ISDN/PSTN target network

BY ARNE ØSTLIE

1 Introduction

This article describes the new logical network structure of Telenor's ISDN/PSTN (noted as *the target network*) and the new routing principles that are introduced.

The work started as early as 1987 in a joint project between Norwegian Telecom Research and the Traffic Engineering Office in the Networks Division. The 7 regions of Telenor have also been active in developing the principles and the network structure (especially Region South participating in calculation of costs for different network structures).

The determination of the actual network (e.g. exchange areas) has been done regionally well co-ordinated with the proposed SDH (Synchronous Digital Hierarchy) structure of the transmission network.

The implementation has taken longer than expected. Still, Telenor is in the forefront in Europe regarding the implementation of the new routing methods and fault tolerant network structure and dimensioning principles.

2 Main principles of the traditional telephone network

In most countries the Public Switched Telephone Network (PSTN) has been based on the same principles, e.g.:

- Hierarchical network based on 4 or 5 national exchange levels. In the case of 5 levels, the 5th is used for international gateways, otherwise one or more national exchanges are combined with international gateways.
- Final circuit groups as a backbone connecting an exchange to its superior in the level above.
- A mesh network of final circuit groups between the exchanges in the upper (4th) national level.
- High usage circuit groups making short cuts in the hierarchical structure of final circuit groups.
- Alternative routing of calls preferring the high usage circuit groups with overflow to final circuit groups.

The structure of Telenor's PSTN has been basically unchanged since the middle of the 1970s. The hierarchical network has 4 levels. The approximate number of exchanges per level before the digitisation started, is given below:

End Offices (EO)
~ 3,000 exchanges

Group Switching Exchanges (GS)
~ 150 exchanges

Transit Exchange Class 3 (FSIII)
~ 50 exchanges

Transit Exchange Class 2 (FSII)
~ 12 exchanges of which 2 – 5 have also been international gateways.

The network connecting the 12 FSII exchanges has been a fully meshed network. Traditionally, only ES and GS exchanges have had subscribers connected. In the transition phase from analogue to digital network, however, subscribers have also been connected to FSIII and FSII exchanges.

At all levels in the network most of the traffic has been carried on high usage circuit groups with overflow to final circuit groups. Figure 1 shows the final route between two EOs in two different FSII regions consisting of seven final circuit groups in series. However, in the example most of the traffic will be carried on a shorter route consisting of two high usage circuit groups and two final circuit groups.

Analogue circuit groups are mostly unidirectional (one way), which implies that the establishment of a connection always takes place from one end. Calls originating from the opposite end have to be set up on another circuit group. However, in the Norwegian digital network bi-directional (two-way) circuit groups are used, reducing the number of circuit groups to one half.

Direct circuit groups have been established when the network cost could be reduced. The optimal size of the direct circuit group has been determined according to *Moe's principle*, which states that the direct circuit group shall be increased until the marginal cost of carrying one erlang extra traffic is the same on the direct *route* as on the final *route*.

For dimensioning purposes traffic measurements have been carried out for 40 days a year according to the *Average Daily Peak Full Hour* (ADPFH) method of ITU-T Rec. E.500. For each circuit group the traffic is measured on an hourly basis. For each day the traffic in the circuit group's busy hour is determined. ITU-T defines the average of the 30 highest ADPFH values of the 365 ADPFH values in one year as *normal traffic load* to be used as reference traffic for dimensioning. Telenor Nett uses similarly the average of the 6 highest of the 40 ADPFH values measured as normal traffic load. Final circuit groups have been dimensioned at 1 % congestion when offered this normal traffic load.

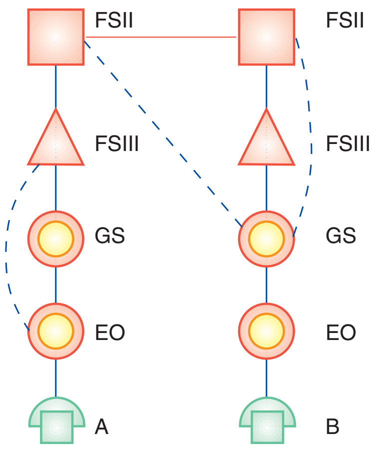


Figure 1 Example of circuit groups between two subscribers A and B in two different FSII regions Traditional hierarchical network

3 Motivation for changing the network structure

In Norway ISDN is considered as an evolution of PSTN. The two services have been fully integrated in *one* network from the introduction of ISDN. This simplifies the capacity planning of the network. However, ISDN and the Intelligent Network (IN) introduce new services in the network which may have higher expectations to quality of service, e.g. availability. The digitisation and technical development is also in other ways changing the preconditions for optimising the network, e.g.:

- The use of concentrators increases the size of end offices. The area covered by a subscriber exchange is therefore no longer limited by the length of the subscriber line. The number of exchanges with subscribers connected will be reduced by a factor of 15.
- The cost of both digital switching and transmission equipment has decreased considerably. Switching costs are reduced by a factor of 10. For digital transmission on fibre cable the infrastructure itself is rather expensive. However, the marginal cost for utilising the fibre cable is low. The trans-

mission cost of large capacities is therefore much less than before.

- Administrative and maintenance costs are no more negligible and favour simplification of network structure and planning and network management routines.
- Increased demand of flexible traffic capacity, e.g. due to TV shopping, IN information suppliers or mail ordering companies which may cause unexpected traffic growth and new traffic peaks.
- Traditional optimisation criteria favour the establishment of high usage circuit groups. However, very often important relevant factors have not been taken into account. Examples of such factors are
  · administration costs
  · underestimated cost effectiveness of final circuit groups due to statistical multiplexing of several traffic streams which also may have different busy hours
  · the utilisation (erlang/circuit) of high usage circuit groups has in practice been much lower than the theoretical optimal utilisation.

  This problem was first discussed by Parviala [1] in 1982 where he argued that analogue high usage circuit groups less than 25 circuits very often were uneconomical.
- Competition will require economic networks as well as improved quality for some of the new services.
- Processor controlled exchanges with signalling system No. 7 make new traffic routing functions possible, e.g.:
  · Load sharing
  · Rerouting
  · Circuit reservation
  · Dynamic routing.

These functions are discussed later in this article.

4 Objectives of the new network structure

The objectives of the new network structure can mainly be described in three words: cost effectiveness, robustness and flexibility.

The cost effectiveness is of course very important in a competitive environment. Keywords in this connection are:

- Administration costs and equipment costs have to be considered together
- Utilisation of existing infrastructure
- Standard and simplified structure
  · reduced number of exchanges
  · reduced number of levels in the network
  · reduced number of high usage circuit groups.

Society is getting more dependent upon the telephone/ISDN services. Therefore, the network has to be more robust against normal, though rare, incidents like

- transmission and exchange failures
- abnormal traffic peaks
- forecast errors
- planning errors.

The market will have greater influence on the development of the network structure. Competition from other services and operators will make the capacity growth uncertain. New products and changed requirements for quality of service, e.g. availability, should be expected. In the future the network should be more flexible concerning

- capacity growth
- changes in traffic patterns
- requirements concerning grade of service and availability which may vary in different parts of the network and for different subscribers and services.

Another objective is that the principles and structure of the logical ISDN/PSTN network have to be co-ordinated with the development of the No. 7 signalling network and the physical/logical SDH transmission network.

Determination of grade of service parameters for e.g. availability is a difficult subject. In this work we have formulated a basic principle in accordance with ITU-T Rec. E.862 "Dependability planning of telecommunication networks":

> *The availability of the network shall be improved until the marginal cost of the improvement is equal to the corresponding marginal value for the subscribers. When selecting between different alternative solutions, the alternative with highest "surplus" should be chosen.*

Or as stated in E.862: "The main object of dependability planning is to find a balance between the customers' needs for dependability and their demand for low costs."

5 New traffic routing functions

For a given destination a routing function in the actual exchange determines which circuit groups may be chosen and the preference between them. In the Norwegian network the call is set up link by link. When a circuit group is chosen, a circuit is reserved for the call and the call set up control is transferred to the exchange at the other end of the circuit group.

In the target network new traffic routing functions like load sharing with mutual overflow, rerouting and Dynamic Alternative Routing (DAR) will generally replace the alternative routing that has been dominant in the traditional hierarchical network. A short description of the routing functions is given below.

5.1 Alternative routing

By *alternative routing* a list common for a set of destinations describes the possible circuit groups in a fixed sequence. A call is routed on the first circuit group in the list which has a free circuit. The last alternative is always a final route. If none of the circuit groups in the list has a free circuit, the call is congested. In practice the list is normally restricted to two or three alternative circuit groups.

5.2 Load sharing with mutual overflow

Load sharing is a traffic routing function which distributes the traffic evenly or according to a fixed proportion between two or more equivalent circuit groups in a load sharing group. One way of implementing the function is to make a random choice according to the given probability. If the selected circuit group is congested, the other alternatives in the load sharing group should be tried.

5.3 Rerouting

Figure 2 shows an example with two independent routes ACB and ADB between A and B. If we have load sharing or normal overflow in A, a call attempt is blocked in A if *both* circuit groups AC and AD are congested. However, if the call reaches C and circuit route CB is congested, there are no other alternatives in C, and the call attempt will be blocked. *Rerouting* (or crankback) is a func-

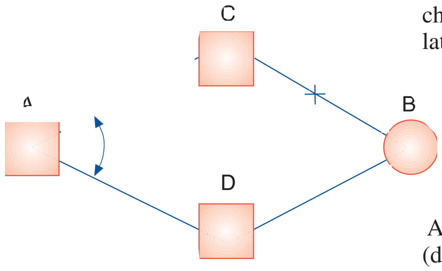


Figure 2 Example of rerouting

tion which returns the call to the previous exchange A. Exchange A can then continue the routing on the next alternative in the routing description, e.g. the other load sharing alternative A-D in the example of Figure 2. According to our specification a call shall not be rerouted more than once in order to reduce the possibility of overloading signalling system No. 7 in failure situations. Similarly, the rerouting function can be turned off manually.

5.4 Circuit reservation

Circuit reservation is a method of giving priority to one ore more traffic types when there is a lack of traffic capacity, or to be more correct: The method determines which traffic type to be congested. The simplest one-level circuit reservation with reservation level R makes all circuits on a circuit group available for one traffic type, while another traffic type is blocked if the number of free circuits is less or equal to R. The simplest functional implementation of circuit reservation is to have a free-circuit counter per circuit group. The counter is counted down when a circuit is seized and counted up when a circuit is released.

The circuit reservation function may be used to

- give preference to priority traffic, e.g. emergency calls or calls from priority subscribers
- give preference to one direction of a bi-directional circuit group
- equalise the grade of service or give downwards preference on a load sharing group
- accept DAR calls only when there is spare capacity to avoid the grade of service for normal calls to be severely reduced.

Some characteristics of the one-level circuit reservation function are described in chapter 7 through calculations and simulations. The function being implemented in the Norwegian network is similar to the two-threshold selective circuit reservation described in CCITT E.412. However, there are 4 traffic types: PRI (priority traffic), DR (direct routed traffic), AR (alternative routed traffic) and DAR (dynamic alternative routed traffic). The two-level circuit reservation may in most practical situations be considered as a combination of two one-level circuit reservation systems. As a result it can be used to combine the abilities of protecting priority traffic, giving preference to down going traffic and rejecting DAR traffic when necessary.

6 Description of the target network

There will be mainly 4 types of exchanges (the expected number of each type in brackets):

- International Exchanges, IE (2)
- Trunk Exchanges, TE (14)
- Local Tandem exchanges, LT (~ 50)
- End Offices, EO (~ 170).

For describing the network some definitions are useful:

- An *EO-area* is defined as the area covered by an end office including connected concentrators and small exchanges (analogue or digital).
- An *LT-region* is made of the EO-areas which use the same LTs for transiting traffic between themselves.
- A *TE-region* is defined as the area where the EO-areas are connected to the same TEs.

EOs and LTs will be subscriber exchanges, except in Oslo where the LTs will carry only transit traffic. The network for national traffic will in principle consist of ***two*** *two-level networks*. For simplicity we will here call the two networks the *LT-network* and the *TE-network*. The two networks are dimensioned separately and built up hierarchically. The LT-network carries traffic *within* one LT-region. The TE-network carries traffic *between* the LT-regions within one TE-region, traffic to/from other TE-regions and traffic to/from the international network.

Figure 3 shows an example of the two two-level networks within one TE-region. In the example the LT-network consists of two LT-regions.

Each EO within an LT-region is connected to the same LTs (two or in a few cases three) with circuit groups in load sharing. This type of connection is often called *double (or triple) homing*. In Figure 3 the thick lines from the LTs symbolise the collection of circuit groups to the EOs.

In a similar way all EOs and LTs in a TE-area will be connected to the same pair (exceptionally triple) of TEs. These exchanges make up one TE-region. Note that the LT only functions as a normal EO (subscriber exchange) towards the TE-network. The circuit groups from an EO to its superior TEs will in load sharing carry all traffic to destinations outside own LT-region. All traffic within the

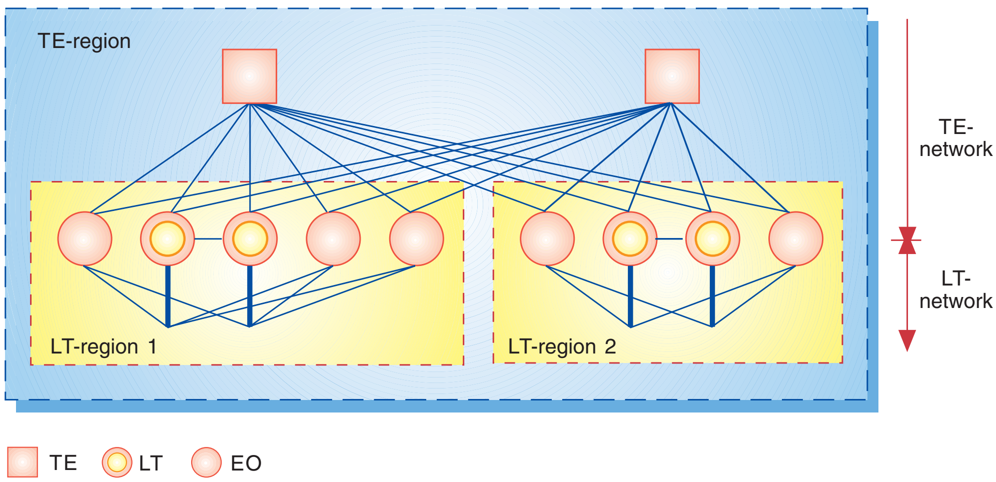


Figure 3 An example showing the TE and LT networks within one TE-region

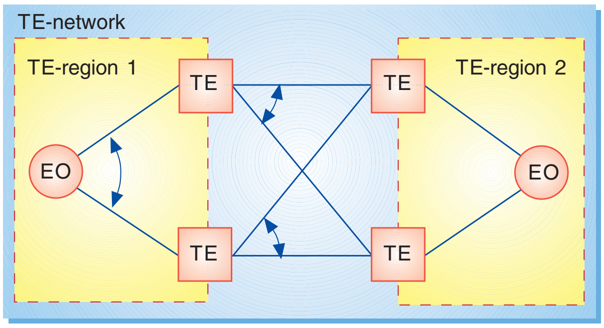


Figure 4 Traffic routing between two TE regions

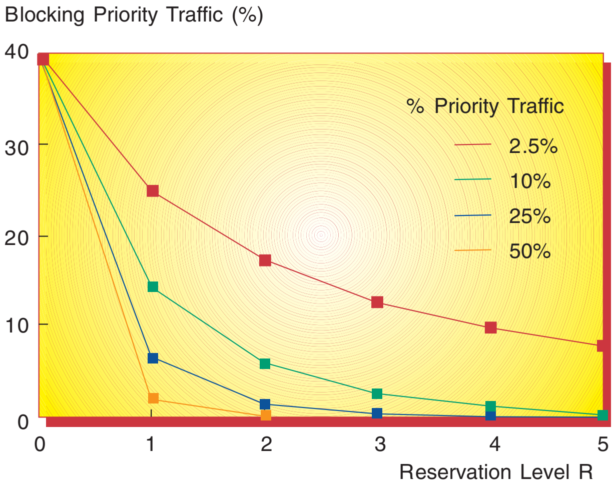


Figure 5 Effect of circuit reservation on the priority stream A_1 during general overload. (B = 40 % when no circuit reservation – R = 0.) N = 60 circuits

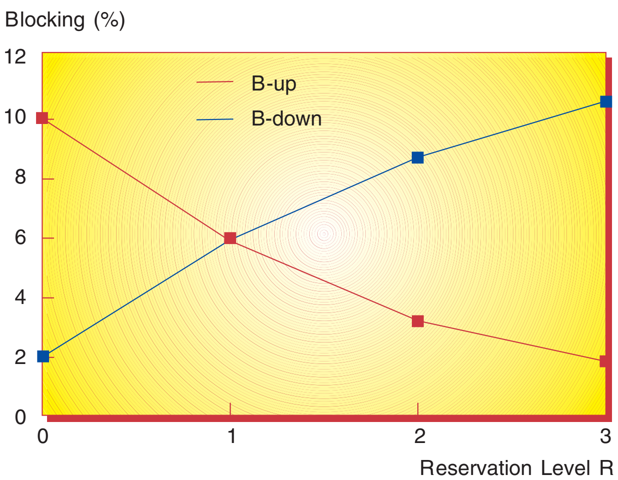


*Figure 6 Circuit reservation for service compensation of double homing. N1 = N2 = 120 circuits. A_{up} = 120 erlang, A_{down} = 2 * 60 erlang*

LT-region will be carried on the circuit groups from the EO to its superior LTs. Thus the LTs have no transit function towards the TE-network.

The 14 TEs and 2 IEs are connected by a fully meshed network. This meshed network together with the access circuit groups EO/LT to TE make up the TE-network. All traffic between two TE-regions (with double homing) will through load sharing be carried on the 4 circuit groups between the two TE pairs as indicated in Figure 4.

The solution with two two-level networks increases the number of circuit groups. However, it has some advantages:

- The LTs are normally large subscriber exchanges with high traffic interests from/to most exchanges in the area. This traffic will be carried in an efficient way, reducing the need for direct routes.
- Most areas in Norway are sparsely populated. The distance EO-TE would therefore be long in many situations. The introduction of LTs makes it possible to keep the number of TEs small.
- The splitting in two networks makes it possible to set individual quality of service criteria to each network, and makes the network less vulnerable.

7 Some properties of circuit reservation

The properties of circuit reservation will be described through some simple examples. Let us consider two traffic streams A_1 and A_2 which are offered the same circuit group with N circuits. The traffic stream A_1 is given priority by reserving the last R free circuits for this traffic. This means that calls of type A_2 are blocked when the number of free circuits is equal to or less than R. When the traffic streams are Poisson processes, e.g. with exponential interarrival times and holding times, the formulas in Appendix 1 can be used to calculate the blocking probabilities for the two traffic streams. If the traffic streams are non-Poisson, e.g. overflow traffic, a simulation model or more advanced analytical models can be used.

7.1 Priority to small traffic streams

Figure 5 shows results calculated from formulas in Appendix 1 when $N = 60$ and high congestion ($B = 40$ % in the case of no reservation ($R = 0$)). The blocking of the priority traffic A_1 is plotted as a function of the reservation level R for different proportions of priority traffic related to total offered load.

This and similar calculations show:

- $R = 1$ gives very good protection for small traffic streams (< 5 – 10 % of normal traffic load). This is true also with much higher overload than in this example.
- $R = 2$ (or $R = 3$) may be considered when the proportion of priority traffic is higher.
- The effect of circuit reservation is the same for small and large circuit groups. This means that the blocking of priority traffic is not very dependent upon the size of the circuit group as long as the proportion of priority traffic and the reservation level are the same and the traffic offered gives the same congestion (when no reservation is used).

7.2 Service compensation for downwards traffic in load sharing groups

On a load sharing group, e.g. from an End Office to the trunk network, an upward call EO – TE will search both circuit groups before it eventually is congested. However, a call downwards from TE will usually be lost if the route is congested. Even if the traffic is evenly distributed on the two trunk routes to EO, the GOS (grade of service) will be better for upwards than downwards calls. The opposite effect is desired: Downward calls near their destination should be given some preference in an overload situation in order to obtain optimal traffic flow.

In Figure 6 are shown simulation results from a load sharing group with two circuit groups, each with 120 circuits. The traffic is symmetric, 120 erlangs up and 2 * 60 erlangs downwards. The figure shows that for $R = 1$ the blocking of the two directions is the same, $B_d = B_u$ = 6 %. Further simulations with symmetric load show that $R = 1$ gives

- some upward preference for low overload, e.g. B_d =1.9 %, B_u = 1.1 %
- downward preference for higher overload, B_d = 35 %, B_u = 56 %.

This is quite in accordance with our objective.

Circuit reservation may also be used partly to protect, partly to compensate the GOS on a final route with overflow from direct routes. R = 2 will be satisfactory, giving some protection of the final route. Several authors have studied this. [2] is recommended.

In the Norwegian network rerouting is planned to ensure that the capacity of both circuit groups is available also for downward traffic. However, circuit reservation may be used in addition to obtain downward preference and reduce the amount of rerouted calls.

7.3 Protection of normal traffic load

The objective of dynamic routing is to utilise the capacity of the network even if the traffic distribution differs from the dimensioned. Two different strategies may be emphasised:

1 The network should be utilised in a way that gives maximum throughput. One consequence could be that a large proportion of a network is affected by a failure situation.

2 Dynamic routed traffic is only allowed as long as it does not severely affect the traffic normally carried on the circuit group. This may be achieved if dynamic routing does not influence the GOS of direct routed traffic more than a fixed percentage.

Dynamic routing in a mesh network may result in instability as two link traffic is competing with direct routed traffic. Circuit reservation against DAR traffic is necessary to avoid this. A reservation level of R = 1/2 SQRT(N), where N is the size of the circuit group, is recommended in order to give maximum throughput in accordance with the first objective above. Simulations by Norwegian Telecom Research [3] support this result. Some calculations using the formulas in Appendix 1 are set up in Figure 7. GOS 1 % at normal traffic is assumed. Also the overload traffic is Poisson traffic in the example. The results show some variation of the blocking of the normal traffic, largest for N = 60 (from 4 to 11 %).

The recommended dimensioning principle in Telenor, *standard dimensioning* (see chapter 9), is more robust towards overload. As shown in Figure 8 this makes it possible to put emphasis on the second objective above by selecting **the same reservation parameter, *R*, for all circuit groups independent of circuit group size**. With our constraints R = 7 limits the blocking of normal traffic to 5 %, similarly R = 10 limits the blocking to about 3 %.

8 Dynamic Alternative Routing

Dynamic Alternative Routing (DAR) is a method that will be used in the top mesh network between TEs as an addition to the direct routing. DAR may be explained by the following example in Figure 9.

A call has entered the mesh network in trunk exchange *A*. Its destination is end office X. From A to X the call is routed as follows:

- 1st choice is always the load sharing group AB/AC. Both circuit groups will be tried before the call eventually overflows to a 3rd alternative, the DAR-alternative.
- The DAR-alternative is determined by a pointer in a circular list of alternatives (DAR-list) exclusive for this traffic destination (TE-region BC). In the example exchange E is the current alternative. This alternative is selected if the circuit groups AE **and either** EB **or** EC have free capacity for DAR traffic (according to the circuit reservation level).
- If the call is blocked on AE, the first leg of the DAR-alternative, the call will be given one more alternative. The DAR-pointer will be stepped to F and the call is offered the circuit group AF.
- If the call is successfully set up to B or C, there is no need to

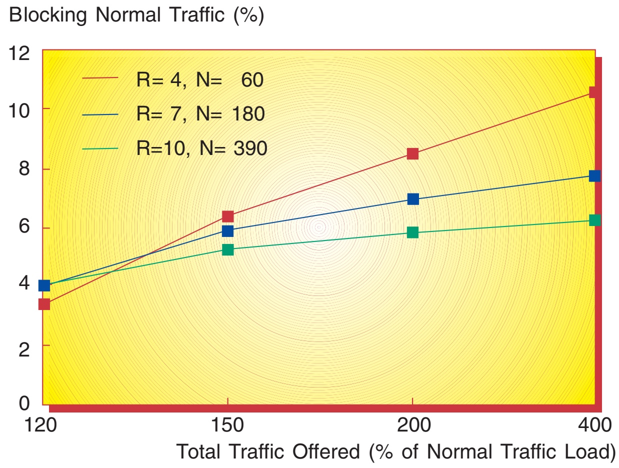


Figure 7 Protection of normal traffic (GOS 1 %) with reservation level R = 1/2 SQRT(N)

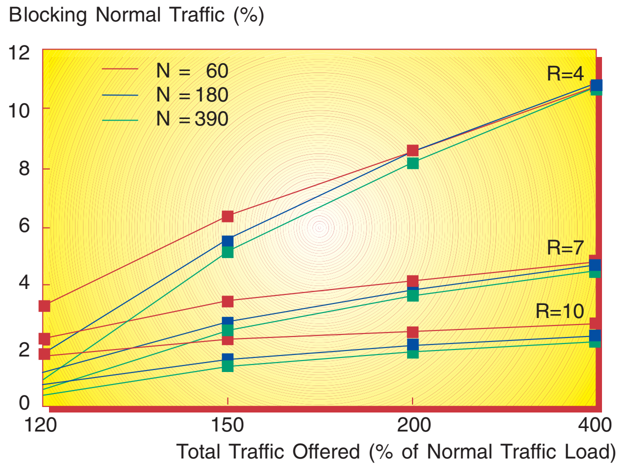


Figure 8 Protection of normal traffic (standard dimensioning) with fixed reservation level

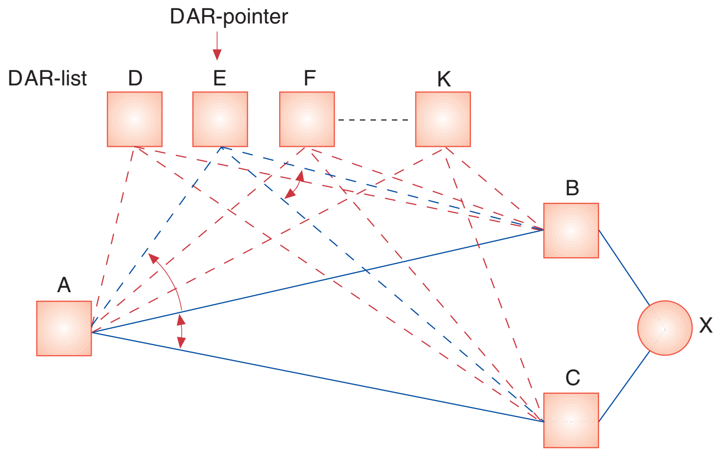


Figure 9 Example of Dynamic Alternative Routing (DAR) in the top level mesh network

change the routing for the next call. The DAR-pointer is not moved.

- If the call attempt is blocked on the second leg of the DAR-alternative, (assuming E is the DAR-alternative, both EB and EC is blocked for DAR-traffic), the call attempt will be lost. A signal is returned to A, stepping the pointer in the DAR list to the next alternative, F. This alternative will be the DAR-alternative for the next call A-X.

DAR is a simple learning algorithm. It was first described by British Telecom. The principle is also used in a method to be implemented by NTT, Japan in 1994. Circuit reservation is mandatory for DAR to obtain network stability and protection of the direct routed traffic. In a simulation study carried out by Norwegian Telecom Research in 1988 [3] our top level mesh network was simulated with different routing strategies under different load conditions like normal traffic load, skew load, focused overload and general overload. The main conclusions were:

- The method gives a flexible network with good traffic flow also when there is considerable difference between traffic actual offered and dimensioning traffic/network resources. The network will be robust towards traffic variations and wrong dimensioning.
- The method distributes the congestion justly.
- The network throughput will not degenerate, but is robust towards general overload. In this case the routing converges towards direct routing.
- The method achieves most of the possible benefits. Compared to more complex adaptive methods, there are no significant differences.
- The method is simple and does not require major data collection from other parts of the network. The additional processor load will be small.
- The dimensioning of the network will not be complicated. The network may be dimensioned as for direct routing.

In [4] G.R. Ash from AT&T presents a comparison between 4 routing strategies

- Learning with random routing (DAR)
- Real time state dependent routing (RSDR – least loaded via route selection based on real-time network status)
- Periodic state dependent routing (least loaded via route selection based on periodic network status)
- Sequential routing with flow optimisation (time-variable engineered routes plus overflow routes updated in real-time (Example DNHR – Dynamic Non-hierarchical Routing).

A large (103 node) and a small (10 node) mesh network were simulated with high day load, global overload, focused overload and cable failure. The basic conclusions were that

- real-time state dependent routing (RSDR) significantly outperforms the other strategies in *large* networks
- DAR and RSDR performs comparably in *small* networks
- all dynamic routing strategies outperform current hierarchical routing by a large margin
- the cost/complexity of DAR may be lower than the other strategies.

As the Norwegian network is a small network these conclusions from 1994 are consistent with our simulation study in 1988.

9 Dimensioning criterion

Telenor Nett distinguishes between

- **Normal load dimensioning** utilising criterion for normal traffic load:
  - *Congestion dimensioning*, maximum congestion at normal traffic load (A_n), e.g. 0.5 or 1 %
  - *Standard dimensioning*, in addition to criteria for congestion dimensioning, a criterion for maximum utilisation (erlang/circuit), U, has to be fulfilled (normally $U = 0.8$). The objective is to make large circuit groups more robust towards overload.
- **Fault tolerant dimensioning** (or reliability dimensioning) utilises criteria both for normal load and failure situation. For failure situations the circuit group is dimensioned to fulfil a defined congestion level at dimensioning traffic during failure, A_f.

  $$A_f = A_n * C_o * C_r$$

  where

  C_o = Overload factor
  = traffic offered the circuit group during failure / normal traffic load on the same circuit group

  Generally $C_o = 2.0$ for dual homing, 1.5 for triple homing. C_o has to be estimated individually for each circuit group in the top level mesh network

  C_r = Traffic reduction factor
  = Dimensioning traffic during failure / Normal traffic load

Some measurements have indicated that in 94 % of the total number of hours in a year, the traffic offered will be less than 80 % of the ITU-T normal traffic load. We have decided to define the network as unavailable when the call blocking is higher than 20 %. As a first attempt our dimensioning criteria were set to ensure that the call blocking in the trunk network should be less than 20 % (B = 20 %) during a failure situation if the traffic offered does not exceed 80 % (C_r = 0.8) of the normal traffic load. The modular dimensioning of circuit groups in addition to yearly network expansion, will, however, give some extra capacity to most circuit groups.

When determining criteria for fault tolerant dimensioning and in which parts of the network the method should be used, the models in ITU-T Rec. E.862, Dependability planning of telecommunications networks, have been very useful.

The use of high usage circuit groups will be rather limited, mainly restricted to circuit groups between large EOs in the same LT-area with high mutual traffic interest.

There are no plans to change the dimensioning criteria when circuit reservation is introduced as long as only a small proportion of the traffic is given priority.

10 Optimal combination of robustness methods

In the conventional PSTN network few systematic methods have been used to obtain a certain level of availability. Automatic restoration in the transmission level is, however, common for the main long distance systems. Also alternative routing in PSTN and occasional use of diversity routing increase the availability.

In the target network the logical and physical (SDH – Synchronous Digital Hierarchy) network structures fit well together. An example of a SDH-ring in an LT-region is shown in Figure 10. In the example the exchanges (EOs and LTs) are connected to the transmission ring by Add and Drop Multiplexors (ADM). Systematic use of double homing will to some extent ensure a level of grade of service also during failure situations. The ring structure of SDH makes it possible to choose between diversity routing in the ring (50 % redundancy) and path protection (100 % redundancy), or a combination giving a resulting transmission capacity K in a failure situation. The grade of service during a failure situation is determined by the dimensioning criteria, the profile of the traffic stream and the transmission capacity K. Failures in transit exchanges will be taken care of. However, EOs and ADMs connecting the EO to the SDH ring, will have no replacement.

Diversity routing may be used either within each circuit group or by routing the two circuit groups from an EO to the LTs opposite directions in the ring. However, the properties will not be the same for all traffic cases. With diversity routing of each circuit group, 50 % of the capacity will be kept on all circuit groups independent of where the fibre cable is cut. If the two circuit groups from an EO to the LTs are routed different ways in the ring and there is a cable fault between EO_A and EO_B in Figure 10, we have the following situation: The circuit groups EO_A - LT_1 and EO_B - LT_2 are not affected by the failure, while EO_A - LT_2 and EO_B - LT_1 are both broken. Therefore, both traffic routes between EO_A and EO_B are broken. If path protection is not fully implemented (or another type of restoration), diversity routing of each circuit group should be recommended.

The optimal choice of robustness level in the physical and logical network is not obvious. In a fault tolerant network an increase in the dimensioning criteria for circuit groups will improve the robustness both against failures in transmission systems and transit exchanges. However, in the LT network, exchange capacity is normally much more expensive than marginal transmission capacity. Therefore, a combination of fault tolerant dimensioning of the circuit groups and path protection of the transmission system may be optimal depending upon the importance of the traffic, the unavailability of the exchange and transmission systems and the actual traffic profile.

Some preliminary investigations utilising the principles of dependability planning of ITU-T Rec. E. 862, indicate that the differences between the traffic profiles of business and residential exchanges (see Figure 11) are important. The residential exchange has a 2-hour traffic peak from 20 to 22 in the evening, while the business exchange has a traffic peak lasting 6–7 hours during the day. As the failure rate is highest during normal working hours and Rec. E. 862 puts more emphasis on business traffic, the following preliminary conclusions are not surprising:

- The traffic profile for *business exchanges* may imply that corresponding circuit groups should be dimensioned with higher capacity (fault tolerant dimensioning). With today's dimensioning level for these circuit groups path protection in SDH-rings seems advisable.
- The short evening peak for *residential exchanges* implies that the proposed dimensioning is sufficient. In many cases no extra path protection of transmission systems is needed in addition to diversity routing.
- Path protection of 60 % of the transmission capacity in addition to 50 % capacity due to diversity routing would result in a transmission capacity during failure of $K = 0.8$. This would be sufficient to cover nearly all single transmission failures with reasonable grade of service. Even 40 % path protection ($K = 0.7$) will give very good robustness. However, the administration costs may favour a solution with full protection or in some cases no path protection at all.

11 Concluding remarks

The PSTN/ISDN target network of Telenor is now under implementation. The network structure with load sharing and dual/triple homing and fault tolerant dimensioning will be completed by the end of 1995. However, not all subscriber exchanges will at that time have two complete independent transmission routes for the circuit groups to the local tandems and trunk exchanges.

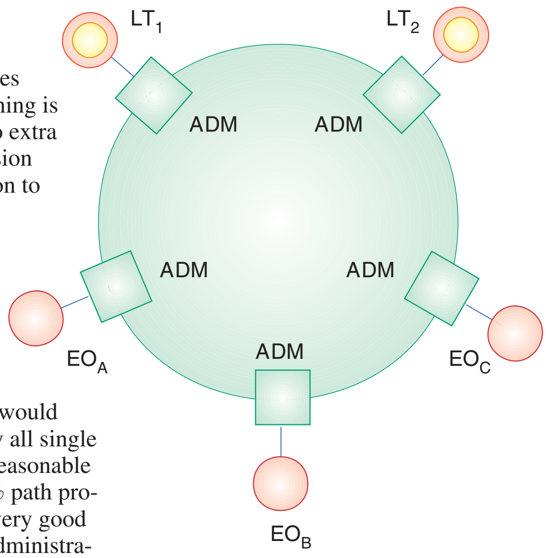


Figure 10 An example of a SDH-ring connecting the exchanges within a LT-region

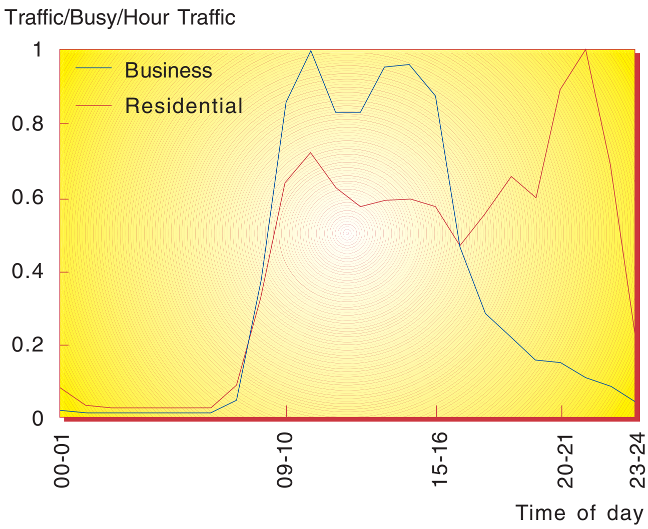


Figure 11 Typical traffic profiles for exchanges with mainly residential or business subscribers

According to our plans rerouting and circuit reservation will be implemented in 1995, while the implementation of Dynamic Alternative Routing is planned for 1996.

At present, the work indicated in chapter 10 is continued in order to make optimal use of protection mechanisms in the physical network and its main user PSTN/ ISDN. A review of the dimensioning rules is part of this work.

As this article indicates, optimal grade of service during failure is a complex subject involving traffic theory, both physical and logical network structure and physical routing as well as traffic routing. A major part of the problem is also the evaluation of lost traffic. However, as the models and the parameters used are not very accurate and the cost ratio work/ equipment is increasing, emphasis should be put on simple models with defined interfaces between the physical and logical network.

References

1 Parviala, A. Optimering av små overflow vior. *Nordic teletraffic seminar,* Helsinki, 1982.

2 Songhurst, D J. Protection against traffic overload in hierarchical networks employing alternative routing. I: *1st international network planning symposium,* Paris, 1980.

3 Pettersen, H. *Dirigering i fjernnettet : en simuleringsstudie.* Kjeller, Norwegian Telecom Research, 1988. (TF-report 53/88.)

4 Ash, G R, Huang, B D. Comparative evaluation of dynamic routing strategies for a worldwide intelligent network. I: *14th international teletraffic congress, ITC 14,* France, 1994.

Appendix 1

A simple model for circuit reservation

We will describe the properties of circuit reservation through a simple model. Two traffic streams A_1 and A_2 are offered a circuit group of N circuits. The traffic stream A_1 is given priority by reserving the last R free circuits for this traffic. This means that call attempts from A_2 are congested when the number of free circuits are equal to or less than R. When the traffic streams are Poisson processes, e.g. with exponential inter arrival times and holding times, the traffic process can be described by the state diagram in Figure A1. The following symbols are used:

- h mean holding time for calls (seizures) of both type 1 and 2 (this simplifies the model, but is not necessary)
- s_1 mean arrival rate for call attempts of type 1
- s_2 mean arrival rate for call attempts of type 2
- s total arrival rate, $s = s_1 + s_2$
- A_1 traffic offered of type 1 ($A_1 = s_1h$)
- A_2 traffic offered of type 2 ($A_2 = s_2h$)
- A total traffic, $A = A_1 + A_2 = sh$
- p_j the probability of being in state j, that means there are j simultaneous calls leaving N-j circuits free.

In the state diagram a transition to the next higher state occurs when a call arrives. When a call terminates, a transition to the next lower state takes place. The arrival process is constant ($s_1 + s_2 = s$) in all states, however calls of type two are congested in the last $R + 1$ states of the diagram, the termination rate (number of calls terminating per time unit) is proportional to the number of ongoing calls.

In the state diagram the number of transitions per time unit between two neighbour states must in average be the same in both directions. Thus, we have the following equations:

$$p_1 1/h = p_0 s \qquad p_1 = p_0 sh = p_0 A$$

$$p_2 2/h = p_1 s \qquad p_2 = p_1 sh/2 = p_0 A^2/2!$$

.

.

$$p_j j/h = p_{j-1} s \qquad p_j = p_0 A^j/j!$$

.

.

$$p_{N-R}(N-R)/h = p_{N-R-1} s \qquad p_{N-R} = p_0 A^{N-R}/(N-R)!$$

$$p_{N-R+1}(N-R+1)/h = p_{N-R} s_1 \qquad p_{N-R+1} = p_0 A^{N-R} A_1/(N-R+1)!$$

.

.

$$p_N N/h = p_{N-1} s_1 \qquad p_N = p_0 A^{N-R} A_1{}^R/N!$$

By using the normalising equation

$$\sum_{j=1}^{N} p_j = 1$$

the state probabilities may be explicitly determined

$$p_0 = 1/\left\{\sum_{j=0}^{N-R}\left(A^j/j!\right) + \sum_{j=1}^{R}\left(A^{N-R}A_1^j/(N-R+j)!\right)\right\}$$

The congestion for traffic stream A_1 is calculated directly as $B_1 = p_N$. The congestion for traffic stream A2 is calculated as

$$B_2 = \sum_{j=N-R}^{N} p_j$$

In the case of no reservation, $R = 0$, the formulas are, as expected, the same as Erlang's 1st formula:

$$B_1 = B_2 = p_N = \left(A^N/N!\right)/\sum_{j=0}^{N}\left(A^j/j!\right)$$

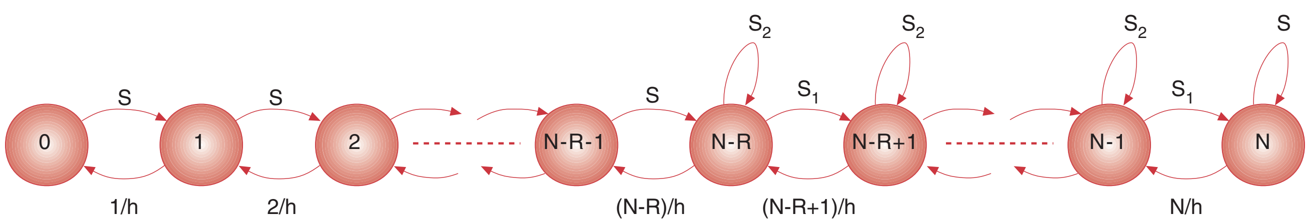


Figure A1 State diagram for one-level circuit reservation with two Poisson arrival processes with negative exponential holding times

Teletraffic analysis of mobile communications systems

BY TERJE JENSEN

As the traffic in mobile telecommunications systems increases the performance analysis from a teletraffic point of view becomes more important. The complexity of such analyses is expected to grow even more when the third generation mobile communications systems will be introduced.

After a brief overview of some aspects in mobile communications systems, a teletraffic model for such systems is presented.

1 Introduction

Systems for mobile telecommunications have been available for several decades. During this time several system generations can be identified, where each generation aims at offering improved service quality and enhanced capabilities, as seen from the users' as well as from the operators' point of view. Such an evolution seems to have the effect on the market that the mobile services become even more popular. This fact, combined with the availability of standards, leads to mass-production of equipment for mobile communications systems. In addition, the competition between operators has been introduced in several countries. The related cost as seen from the users has mostly decreased when related to other living expenses. Naturally, compared with wireline services the wireless solutions allow the users to be more flexible when utilising the telecommunications services. The combination of these factors seems to attract more users. In fact, as seen today in several countries, the growth in the number of subscribers in wireless communications systems is higher than the growth seen in the fixed networks (wireline connections).

Furthermore, a number of telecommunications services are expected to be supported by future wireless systems. It is assumed that the users will request most of the services that are supported in the wireless network. On the other hand, the radio frequency spectrum available for mobile communications services is limited. Therefore, the growth in the number of users, if it is accompanied by a corresponding growth in the offered traffic to the system, results in a need for new solutions for the system structure. In addition, a demand is expected for using the same mobile terminal in most of the environments, like at home, in the office, in the streets, and so forth. Other types of terminals could be used for special services. The phrase "any place, any time, any form" is often seen in the discussion of future systems.

The result may be that a number of operators can provide the same service as seen from a mobile station at a certain location. The various providers can have different purposes for covering an area, e.g. a company would implement wireless services to its employees, while a telecom operator provides the services for the public. Which mobile station that can utilise the various operators' equipment could be controlled by regulation rules and authorisations.

Such situations, together with the increased traffic demand, are expected to lead to a mixture of coverage areas for the base stations (usually named cells). That is, some base stations cover large areas while others cover smaller areas.

There seems to be an emerging interest for such solutions, as they give the operators more flexibility in the way they provide the service in an area. However, an increase in the complexity will also accompany such structures.

As seen from an operator, the primary interest can be to increase the traffic handling capacity of a mobile communications system. This is often measured in the number of user connections that can be handled (e.g. a variant of the Erlang measure). There are also several other measures for the performance of a mobile telecommunications system.

A question that arises is how to model and how to analyse such mixed cell structures for the estimation of some teletraffic variables. These tasks can be relevant in several of the system life phases, e.g. during the pre-standardisation, standardisation, equipment design and network dimensioning/service deployment.

One of the objectives of this paper is to give a brief introduction to what influences the performance of such systems. Another objective is to present a framework for models and corresponding analyses of mixed cell structures.

In the following, a review of the wireless systems will be given as they evolve in time and for the various application areas. Then, some of the functionality will be described and the performance questions discussed. In the second half of the paper, a model for future mobile telecommunications systems will be presented and the main steps in the analysis are outlined. One example is included. A list of the abbreviations used is given at the end.

2 Mobile telecommunications systems

Before going into the functionality and the system performance topics, a somewhat broader view of mobile communications systems will be taken on.

2.1 A historical perspective

Just after H.R. Hertz' discovery toward the end to the nineteenth century, the possibility of transmitting speech by the use of telephones without wires was recognised [3]. However, this was more like a conjecture at that time. The first experiments with radio on land communicating between moving radio units seem to have been carried out in 1921 in Detroit, USA. This was a broadcast service (one-way) and followed by experiments with two-way systems that were conducted during the 1930s. However, during World War II the need for such systems on a larger scale was pronounced with more weight. One purpose was to control the operations of the military units. After the war the work on improving the services was intensified, including the possibility to establish connections between a mobile radio unit and a terminal in the fixed network. It seems that the public market was not waiting for these systems and the demand was limited. This may explain the reluctance of introducing new public mobile communications services in the 1960s and the 1970s.

The cellular concept (including handover capabilities) was proposed in the early 1970s, but not utilised before the automatic systems were introduced in the beginning of the 1980s (late 1970s in Japan). Before this, manually operated systems had been in service for several years. If we define the cellular concept by the reuse of frequency in distant cells, the first idea of this seems to have been presented in the last part of the 1940s.

The Nordic Mobile Telephone (NMT) system became available in 1981 and resulted in a growth in the number of users which exceeded most of the prognoses for the countries where it was introduced. For larger cities the capacity soon limited the number of users and a system in a higher frequency band was designed. This was also experienced in USA where the first commercial cellular system was introduced in 1983, and by 1984 some cells were reported to be saturated.

Based on the experiences with the analogue NMT system the work on describing a digital system, Global System for Mobile communications (GSM), began. Before any digital system was ready, a number of analogue systems had been introduced in Europe, resulting in a range of incompatible systems. From a European perspective one of the purposes of the digital system was to get a common system in order to gain from the mass production and compatibility between the different countries. By the introduction of GSM in Europe competition between mobile network operators was also realised for the first time in many of these countries. The growth in the number of users experienced in several countries makes the mobile communications systems one of the fastest expanding fields in the area of telecommunications. However, the actual rate differs, e.g. due to the fact that alternative systems are available in some places.

Cost and service quality (including coverage) are important factors in this context. In general, these aspects are not specific for mobile communications systems. Increasing awareness of the market and focus on the possibilities are also among the fundamentals in order to arrive at high penetration rates.

Besides the public communications systems, the use of private mobile radio systems and cordless systems have also been growing. Based on the expected demands (number of users and traffic) the work of defining future system(s) has been initiated.

As the traffic load for some areas saturated the system, solutions like allowing for additional frequency spectrum and introductions of new systems were tried. Usually, the functionality incorporated in those systems did not allow for more sophisticated distribution of the traffic load. In that sense the traffic modelling problems were limited in their complexity.

2.2 Generations of wireless systems

The various wireless systems can be arranged into a number of generations as shown in Figure 1. Also depicted in that drawing, several application areas of such systems can be defined. This will be returned to in the next section. In addition, several manually operated systems were in use before the first generation of automatic systems were introduced. Systems intended for specific markets, like paging, mobile data and private mobile radio, are not included in this presentation.

Although there are differences between the systems, some commonalities can be mentioned. Perhaps, the most pronounced one is that analogue systems are found in the first generation while the systems belonging to the second generation are digital. Another difference is the use of Frequency Division Multiple Access (FDMA) for systems in the first generation and Time Division Multiple Access (TDMA) for those in the second generation. The use of Code Division Multiple Access (CDMA) is examined as an additional candidate for the latter systems.

The systems defined in generations later than the second are still under specification. Especially in USA, work is carried out to describe a concept named Personal Communications Systems (PCS) which is often said to belong to generation 2.5. Which multiple access schemes to use for the systems in the third generations is only one of the questions that must be answered during the work with the standards.

There are mainly two arenas where standardisation of Third Generations Mobile Systems (TGMS) are carried out. One work scene is the study groups in ITU (International Telecommunication Union), in special Task Group 8/1, in the Radio Communication Sector. They adopted the working title Future Public Land Mobile Telecommunication Systems (FPLMTS). In Europe we find UMTS (Universal Mobile Telecommunications System) used as the working title of the concept standardised by ETSI (European Telecommunications Standards Institute), sub-technical committee SMG 5.

Although different organisations are involved and different names are used, it is foreseen that UMTS may be based on or be identical to the world-wide standard (FPLMTS). That is, compatible solutions could evolve.

2.3 Application areas and services

Several application areas of wireless systems can be defined. Some are given along the horizontal axis in Figure 1. In addition, other classifications have been identified, several illustrated in Figure 2. Four environment classes could be mentioned:

- domestic
- business
- mobile/vehicle
- public.

These could be further detailed by the service demand, i.e. as given by the measure Equivalent Telephone Erlang, ETE,

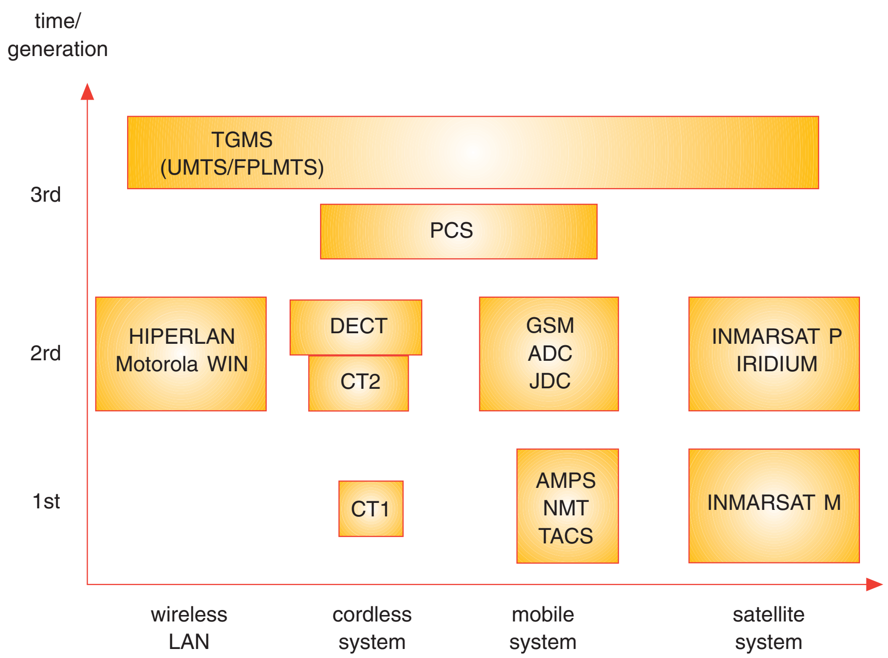


Figure 1 The development of wireless systems. The systems depicted are only examples as more systems have been defined

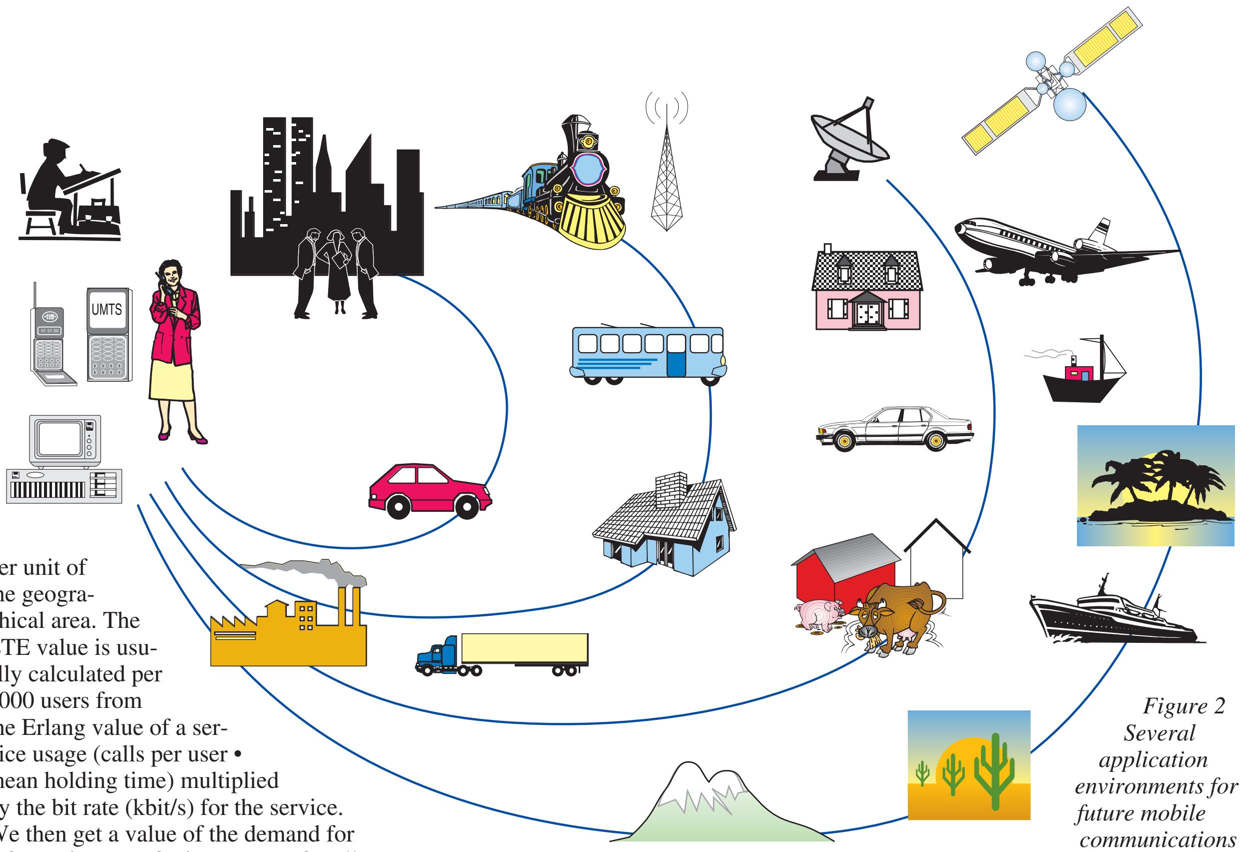


Figure 2 Several application environments for future mobile communications systems

per unit of the geographical area. The ETE value is usually calculated per 1000 users from the Erlang value of a service usage (calls per user • mean holding time) multiplied by the bit rate (kbit/s) for the service. We then get a value of the demand for information transfer in common for all the services.

Several classes of operators can be defined as well, like private, public, and residential. Private operators will typically be companies handling their own employees, while a residential operator handles a few users in the residence that often are associated with a main user.

For a TGMS a strive for minimising the number of radio interfaces and definition of blocks related to the radio interface that could be adapted to the situation experienced are carried out. This invites for the use of a layered and building block approach to allow for such a flexibility. Comparing a TGMS with the present generations of systems, we recognise that the latter have more or less been designed for a limited number of services and application areas. Therefore, the parameters on the radio interface could be tuned correspondingly.

Some of the services that could be supported in a TGMS are depicted in Figure 3. Information bit rates potential up to 2 Mbit/s have been proposed. In addition, studies are going on for even higher bit rates, e.g. for mobile broadcasting.

We could compare this with the bit rate supported in GSM, 13 kbit/s, and in DECT, 32 kbit/s (although several slots could be used for one connection). Naturally, the highest bit rates will not be supported in all the environments by a TGMS. The highest bit rates may be achieved for indoor connections while some hundreds of kbit/s may be relevant for outdoor over short distances (mobile stations close to the base station). For longer distances around 100 kbit/s could be a limit for the information rate. However, all these figures remain to be verified for real cases.

3 Functionality in mobile communications systems

The functionality present in a system can be described in two steps: first, the various sections, and second, the layers within each of the sections. We may also look upon this as a division between a horizontal and a vertical description of such systems.

3.1 Sections of a system

In principle, mobile telecommunications systems have more or less the same functional architecture for several system generations, as illustrated in Figure 4. However, the physical entities, protocols, etc., that are included do vary, in addition to the mapping from the functional entities to the physical elements. Starting from the users' side, each user has an identity, associated with a terminal (as a fixed terminal identity or as a separate unit, e.g. a Subscriber Identity Module, SIM, in GSM). From GSM and onwards, the subscription will be linked with the user, i.e. according to the described role model, and not to the terminal. Then, a user could have an access device to ease the identification and authentication procedures.

A mobile station includes the terminal equipment and the mobile termination (MT). The terminal equipment could also be more complex than a telephone device, e.g. like a PABX. In the systems available so far, most mobile stations consist of one physical unit. Such units decrease in size and weight with the development of technology. In addition, some of them have the possibility to be interconnected to standard terminal equipment.

When a communication (information transfer) need arises, a relationship between the mobile station and a base station is established. Such a need could be initiated by the user or as a result of the invocation of another type of procedures defined in the system.

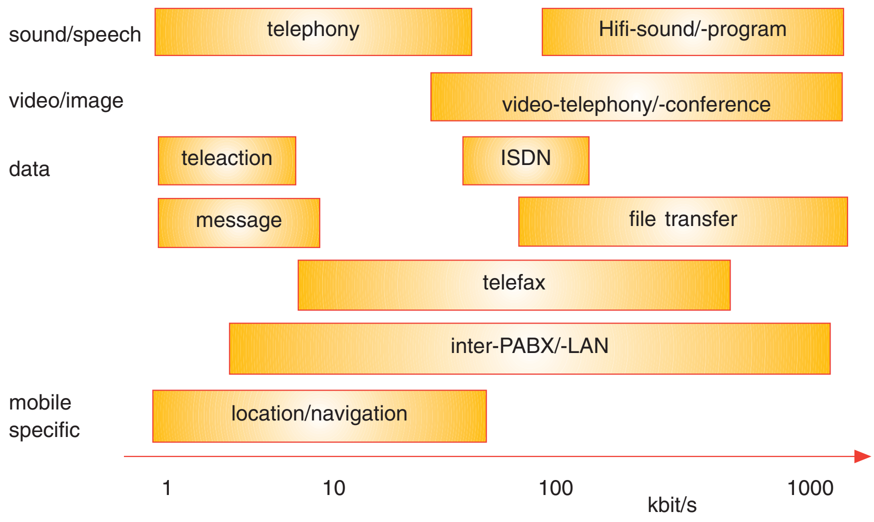


Figure 3 Some services that might be supported in a TGMS

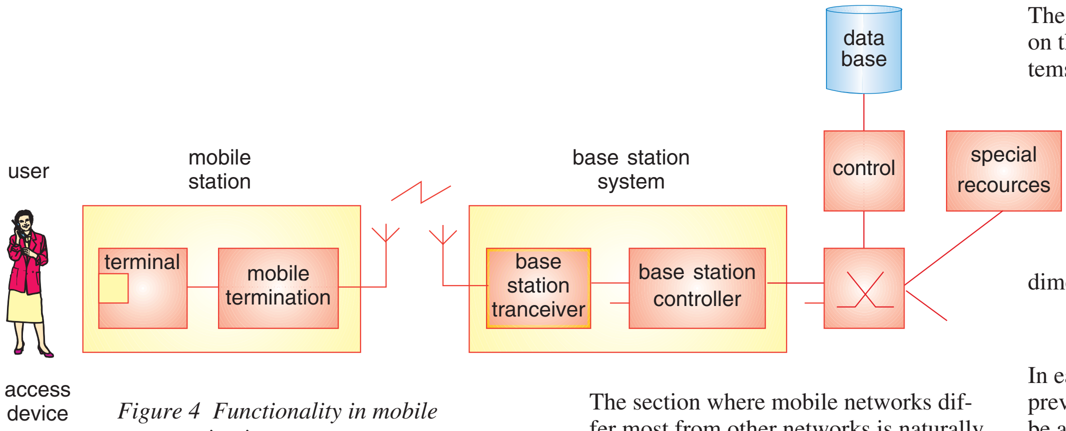


Figure 4 Functionality in mobile communications systems

A base station system can be composed of a transceiver and a controller. A transceiver consists of the transmitter and the receiver units for (de-)multiplexing, frequency conversion, channel (de-)coding, etc. A number of transceivers could be controlled by a base station controller that may perform channel allocation, some cryptographic procedures, etc. A switching function is normally a necessity in any larger system. Control and data functions are required for handling the mobility, the services and the storage of related information. Following the architecture of an Intelligent Network (IN) special resources could also be defined implementing voice mail, answering machines, etc.

Even though the functional entities are depicted as separate units, the physical implementation may vary, e.g. by connecting the elements with a MAN (Metropolitan Area Network), ref. e.g. [5], [9].

The section where mobile networks differ most from other networks is naturally the air interface, i.e. between the mobile stations and the base stations, together with the corresponding functional units. For the other sections, we will find tasks similar to those valid for most telecommunications systems. These include both the handling of user connections and the signalling. However, specific considerations due to the mobile nature of the users/terminals must be included.

An additional aspect is that several operators/providers may be involved, requiring mechanisms for mutual recognition of identities and rights.

As we find a high level of resemblance between the wireless and fixed systems, the reuse of functions and the integration are topics of special interest. For integrating a mobile system and a fixed network, a number of integration levels can be defined, e.g.:

- service
- signalling
- functional.

Here, integration means common use of the relevant elements. For instance, by signalling integration the signalling messages and the procedures are capable of dealing with the requirements from each of the systems (mobile and fixed terminals).

Two extreme integration schemes can be described: firstly as a stand-alone solution and secondly as a completely integrated solution. The first case could be requested to ease the introduction of mobile systems. The latter solution could be the most economic one according to the scope considerations. However, some enhancements of capabilities in the fixed network may be needed for the fully integrated solution, e.g. to allow for unrestricted handovers between the various application areas.

The integration solution also has impact on the performance studies of these systems. If the equipment examined is used for handling a number of services, the load impact from these services should be included in the study. Most likely, this would increase the dimensionality of the problem, at least before applying techniques for reducing the dimension.

3.2 Layers

In each of the sections described in the previous part, a number of protocols can be applied, depending on the integration scheme chosen. For each of the protocols certain aspects could be studied. Some protocol stacks, as defined in international standardisation organisations, are usually elaborated. The number of layers defined will depend on the configuration, e.g. a single link, a network, and the efficiency to be achieved, e.g. to avoid some of the overhead information. In addition, each top level protocol can be aimed for controlling a few of the aspects relevant for the system. For instance, one relationship can be defined between the base station controller and the mobile control function, while another is defined between the mobile station and the control function. Naturally, these would have different purposes. When the performance of a link with related protocol processors are examined, the characteristics of each of the protocols could have an impact on the resulting performance measures.

In the physical layer (according to the protocol stack) several of the mechanisms present will influence the resulting quality of a connection. Some of these mechanisms are:

- service/source coding
- interleaving
- physical level encryption
- channel coding
- burst building/formatting (duplexing scheme)
- modulation
- multiplexing
- power control.

Most of these mechanisms must be harmonised both in the transmitter and the receiver side of the air interface. When examining the capabilities of these mechanisms in combination, the overall system performance and the performance as seen from each user should be considered, as will be returned to in Section 4.1. In addition, a number of other features can be identified, like the equalisation and the diversity scheme.

Error control techniques can be classified as forward error control or automatic repeat request. Some performance measures relevant for error control are the throughput (and goodput), integrity, and delay. In this context, goodput is the rate of error free information (user level) that is transferred. Naturally, automatic repeat request may introduce long delays, especially when the radio signal conditions are poor. This fact can be compared with the more constant delay that forward error control will introduce. Another aspect is that the automatic repeat request is more likely to keep the error control within a given limit. These capabilities make the different error control techniques suitable for various services, like automatic repeat request for data services and forward error control for services having real time requirements.

Returning to the portion specific for mobile systems (the radio interface), several channels are usually defined, as depicted in Figure 5. Several of these channel types have different usage patterns and ways of control. Some are one-way in the sense that the information flows either to or from the base station, e.g. like the broadcast control channel usually carrying information from the base station to the mobile stations. The structure of these channels has so far been fixed during the system specification phase.

Various performance characteristics can be relevant for the different channels. For instance, when one entity (e.g. the base station) controls the information transmission, the maximum transfer rate and delay could be of most interest. When several entities access the channel, stability considerations should be included as well.

One observation from the preceding discussion is that the different mechanisms should be dynamically adjusted for the various services and the environments, see e.g. [1], [4]. Utilising such a flexibility will support the introduction of TGMS into the application areas as the need for an overall trade-off between the various mechanisms could be alleviated. However, the complexity will increase, implying a change of the cost, which then invites for the definition of other trade-off problems as well.

3.3 Mobility procedures

In addition to the usual call handling procedures (call set-up, call release, authentication, supplementary services, etc.), a number of more or less mobile specific procedures can be defined. Such procedures mostly result from the fact that the users are moving. They include:

- handover (switching an established connection between channels in different base stations or within a base station)
- location update (performed when a user/terminal is crossing a location area boundary, which can be defined by a set of base station coverage areas)
- paging (searching for a mobile)
- registration/deregistration (linking/delinking a user to/from a certain terminal)
- attachment/detachment (informing the network of a terminal/user status, e.g. power on/off for a mobile terminal).

A number of trade-off problems (optimisation problem formulations) can be identified, several related to the division of the coverage area and the decisions of which criteria to use for initiating the relevant procedures.

A location area can be defined as the area within which a mobile station can roam without informing the fixed network about its whereabouts. Entering a new location area, the mobile station sends a location update (registration) to the network. When a call for a mobile station arrives, the network has to search for (named paging) the mobile station in the location area where it was last registered.

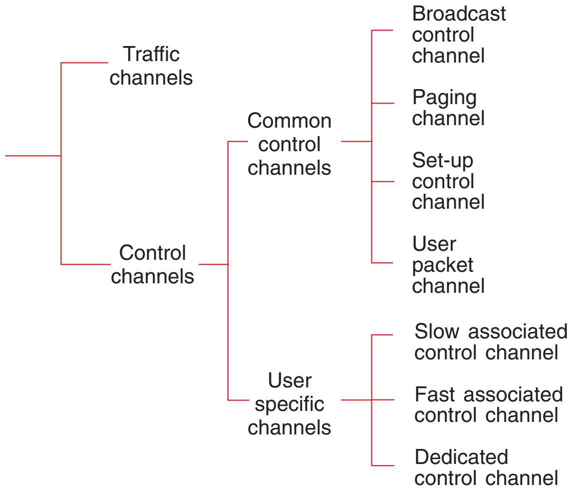


Figure 5 Categories of some possible channels in a mobile communications system

In a network with cell sizes of the same range and without overlap, minimising the location update traffic requests large location areas, while minimising the paging traffic requests small location areas. By a suitable weighing between location updates and paging messages, this can be formulated as an optimisation problem, see e.g. [12]. Such problems are expected to become more complex in multi-layer cellular networks.

The shape and the location of a cell may have a major impact on the handover traffic, e.g. as shown in [10]. This effect is more pronounced for smaller cells where the distinct paths (routes for mobile stations), their orientation and traffic flows become important. For a larger cell the approximation that the moving directions of the mobile stations are uniformly distributed over (0,2π) could be more justified. Some possible distributions for the dwelling times in cells are studied in [6]. Under suitable approximations (ned total call duration, random change of direction within a cell, etc.), it was found that a ned cell dwelling time can be assumed. As also pointed out, it is doubtful whether this result is valid for smaller cells. Then, any regularity in the mobile stations' activity patterns should be considered.

The influence from the cell geometry on the handover rates together with the grouping of cells and the location update rates can also be studied. The routes followed by the mobile stations could be taken into account when planning the base stations' coverage areas. Another point is to introduce hystereses for the decisions when a handover should be initiated in order to avoid multiple handovers for mobile stations that move along the border of two cells.

3.4 Increasing the capacity

In this section, system capacity means the amount of traffic that a system is capable of serving. Most systems of today are interference limited in the sense that the level of the interfering signals will decide the number of calls that can be connected. In order to increase the system capacity, the level of the interfering signals should be reduced. This can be done by a more directed delivery of the radio signal power from the transmitter (e.g. the base stations) to the receiver (e.g. the mobile station) while no radio signal is transmitted when it is not needed. One example is the utilisation of smaller cells for base stations and the use of sectorized cells. The result can be that a lower power is transmitted and a closer reuse of the frequency (measured in geographical distance) can be achieved. For areas with high traffic demand such solutions are relevant. Therefore, a tempting approach could be to reduce the coverage area for each base station in order to increase the reuse factor of the radio capacity as measured per geographical area, e.g. per square kilometre. However, this could become an expensive solution as the cost of the base station network in principle is related to the inverse of the square of the cell radius. In addition, it could be difficult to cover an area completely with such smaller cells.

However, as areas with a lower traffic demand will remain, future wireless systems must allow for larger coverage areas as well. Another consideration is that mobile stations usually should be connected to a base station for a minimum time. As the coverage areas are decreased, mobile stations would move relatively faster (as seen from a base station). This results in shorter handling times in each base station and increases of the signalling load for control of the connections for these mobile stations.

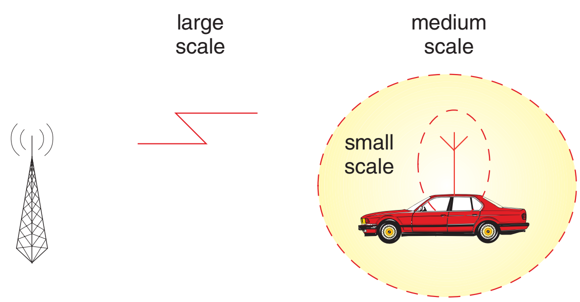


Figure 6 Three scales could be studied when examining the radio signal propagation

One solution can be to introduce a mixture of coverage areas for the base stations, in a range from smaller to larger ones. The velocities of the mobile stations served will also influence the size of the coverage areas.

In addition to the propagation/interference considerations, we should provide the capacity to the areas where it is most requested. If fixed capacity allocation to every base station is applied, the utilisation of smaller coverage areas could make the situation worse if the traffic demand is not according to what was assumed during the allocation process.

Another aspect is the change of the traffic demand during the day and during the week. It is expected that this will have a higher variability for future systems as the number of users increases. For instance, a football match could gather several thousands of persons. Some events during the arrangement could trigger several of the spectators to make calls. During the rest of the week this football arena could be more or less empty with a very low traffic demand.

In order to deal with such varying traffic demand a hybrid capacity allocation scheme could be applied. In this context, the term hybrid includes both the fixed allocation and the dynamic allocation as the two extremes. When a fully dynamic scheme is utilised no capacity is fixed to any of the base stations.

For a future wireless system with various base stations (hierarchy), overlapping coverage areas and a number of operators, a fully dynamic scheme might be difficult to implement. Another fact is that were such a scheme applied, most of the capacity would be allocated to the base stations that were the first ones to be tried by the mobile stations. That is, the other base stations could suffer from this.

Although a fully dynamic scheme could be possible in principle, some maximal limits for the capacity allocated to a base station will most likely be present in a real case. For instance, the number of transceiver units, capacity on the fixed line side of the base stations, etc., would have to be decided for most cases.

In view of the presence of different and overlapping coverage areas and differing mobile stations' service usages, the situation for a hybrid capacity allocation in a future wireless system is more complicated. In relation to the implementation of hybrid allocation schemes, there are several decisions that have to be made regarding the algorithms and the sets of criteria that can be used, see e.g. [7].

4 Some performance topics

From the outset we can examine several measures as the quality of a wireless system. We may also think of modifying the relevant system mechanisms in order to improve that specific performance measure. However, it is usually the overall system performance that is of most interest. In addition, we may see this overall performance from the users' and from the operators' points of view.

4.1 Performance measures

Three mutually independent and multiplicative radio signal propagation phenomena seem to be relevant, see Figure 6:

- large scale signal loss according to a relevant model, like free space, earth reflection, and diffraction
- medium scale, e.g. because of shadowing, often modelled as log normal distributed loss
- small scale for multipath fading due to scatters and reflections of the radio signal.

The small scale is related to some fractions of wave lengths while the medium scale may concern some tens to hundreds of wave lengths. For a signal with a frequency of 1.8 GHz the wave length is approximately 17 cm.

Propagation studies can be divided into the examination of two different characteristics: the multipath propagation characteristics and the path loss characteristics. The multipath characteristics will influence the maximum possible bit rates and state some requirements to the channel coding/equaliser.

Several studies of radio propagation have been presented. Many of the results indicate that the propagation loss is not inversely proportional to some fixed power of the distance (slope). Often, the loss is inversely proportional to a slope that varies, depending on the distance from the transmitter. The main cause for this variation seems to be the multipath propagation. The slope depends on the environment, e.g. the number of scattering objects and the direct link condition.

Several path loss models have been proposed, both empirically/statistically and theoretically (deterministically, numerically) based. In empirical models data on building masses, terrain profile, etc., are often used, while in the theoretical models we could calculate the conditions by applying a number of discrete rays.

Several measures could be used when describing the radio coverage quality. Some examples are:

- the probability of outages due to fading
- the length of an outage period, and
- the bit error rate.

As future wireless systems will be applied indoors and outdoors, indoor propagation as well as the wall penetration are of interest.

The system capacity is usually referred to as the capability of handling traffic channels. Considering that digital systems are used, one possible measure is a value specified in the number of kbit/s/Hz. However, in order to take into account the spatial effect, kbit/Hz/km^2 could be applied. Then, the mean reuse distance will be included as well.

If speech is the main service, kbit/s could be substituted with the number of connections established. Another way is to give all the service demand in ETE as described above.

Then, these and similar measures could be used for comparing different systems and techniques. Another kind of measures, those related to the cost of providing a service, will not be dealt with here. However, the economics of scope by utilising common components for mobile services and other services (fixed network) should be further examined. Related to this, the complexity of the units in the mobile communications systems could be included in order to estimate the cost of the resulting system implementation.

In addition to the radio coverage, other studies separated both in the section and the layer sense can be carried out. For example, each of the channel types depicted in Figure 5 can be looked into with regard to the delay and blocking. The main variable in the rest of this paper is the blocking probabilities for user specific channels on the radio section.

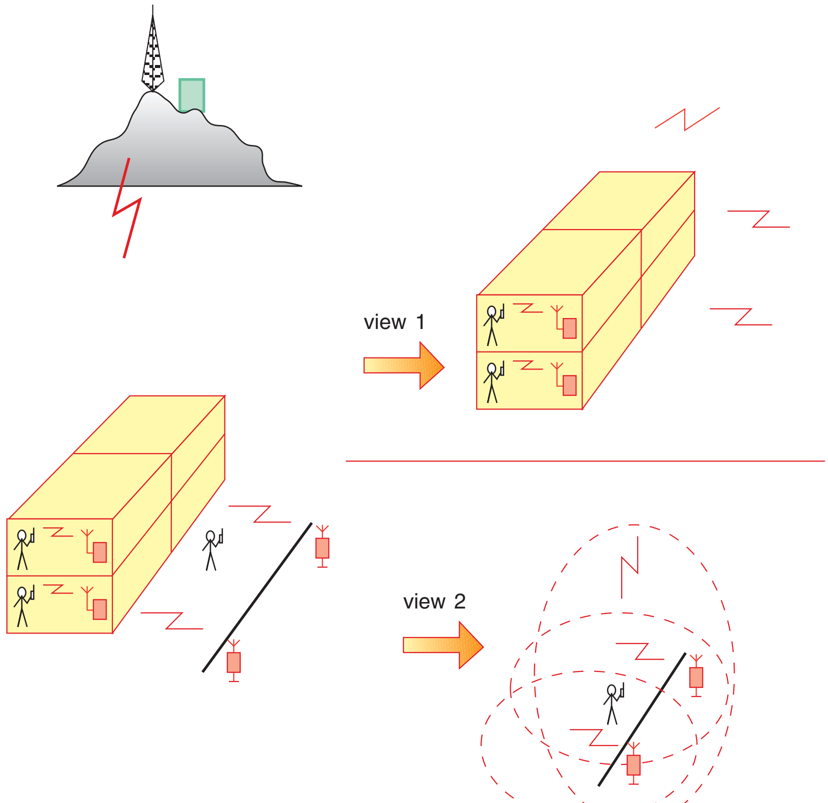


Figure 7 Several views can be selected corresponding to what we want to examine

4.2 Teletraffic model

4.2.1 Modelling goals

Some factors describing a TGMS could be summarised by the terms multi-environments, multi-operators, multi-services, and multi-classes of mobile stations. Naturally, it can be questioned if we should handle the possible combinations (and aspects) in one model or if a number of models should be identified. An advantage of having a set of models is that each can be "tailored" to its purpose. A drawback is that it could be more difficult to handle the dependencies between the various modelling views. The other approach, which will be the one pursued here, is to establish a modelling framework that could be applied for most of the situations. A challenge is to allow the flexibility foreseen for a TGMS with only one basic model. The reward is that the analysts could learn only one model with respect to the necessary input and the results. However, several decisions still have to be made when performing the analysis implying that the one-model solution may not be all that simple.

The overall goal is to establish a simple and intuitive model for a future mobile communications system which estimates the interesting variables with a reasonable level of accuracy. Such a model must be simple in order to describe larger areas, although the area could be treated in several parts. The aspects of the model should also reflect physical phenomena.

The level of accuracy should correspond to the application of it, e.g. if the model is used when comparing different configurations, only relative changes may be of interest. If absolute measures are requested, a higher level of accuracy might be required. In addition, the level of accuracy should consider how accurate the input data can be estimated. If much work is laid down in order to give the input data with a high level of accuracy we usually want the results (output) to be given with a corresponding high level of accuracy.

We also want to select the level of detail for modelling the relevant area, see Figure 7. That is, we may choose to neglect details when larger areas are studied and only more coarse characteristics are sought. On the other hand, more details can be considered if a smaller area is studied and fine characteristics are relevant. Details of the radio propagation might be neglected when large scale effects are examined. Illustrated in Figure 7, in the first view we analyse the situation inside a building. In the second view a street situation is studied. Most likely the first view would consider a smaller area compared to the street situation. Therefore, more details may be included.

In the same way we want the model to adapt to the scale. For instance, we could choose to regard some phenomena in more detail while others could be lumped together or considered as outside the modelled part. In this way base stations covering small areas could, for instance, be considered to be outside the model. Then, any handover calls from these base stations to the base stations included in the model could be regarded as new calls with a modified call duration. Several classes of mobile stations could also be regarded as one class while studying a particular class in more detail. Figure 7 shows some examples. In the first view we have chosen to consider the outside base stations as interference sources, another solution is to include them as secondary servers, i.e. they would handle any overflow traffic for some services. In view 2 the base stations inside the building are not taken into account explicitly. In the more general case, a part of the area could be modelled with a greater detail while other parts are represented in a more coarse way. In addition, we could use more coarse models in order to get some overall results quickly.

The requested output data or results we want to get from such a model are also included in the requirements to the model. They can be divided into two categories, see Figure 8. In the first category we find variables related to how the user/mobile station experiences the services. Here, these are commonly named service quality variables. Some service quality variables are:

- blocking probability for new calls
- probability that a handover will be blocked
- probability that a call will be released due to blocking of a handover request
- probability that a certain base station handles the call as seen from a given location (may influence the quality of the connection as seen from the user and could be related to other variables like the outage probability).

The second type of variables are those used for the dimensioning of the fixed network. Examples of such variables are the call arrival rates and the service times for a base station. These will give requirements to the call handling capacity related to that base station, e.g. processing capacity, number of transceivers and the capacity of trunks and signalling links connected to the base station. The amount of traffic served by a base station is usually closely related to the income (and charges) of the calls using that base station and the accounting between the operators/providers.

Finding the handover traffic between any pair of base stations is also requested as this traffic implies a load on the signalling network as well as load for some elements representing the functional entities in the fixed network. The delay requirements for handover procedures are expected to be more pronounced as smaller cells are involved (relative to the velocities of the mobile stations). The combination of the handover traffic and the related procedures would then be useful when studying alternative architectures for the structures of base station interconnections.

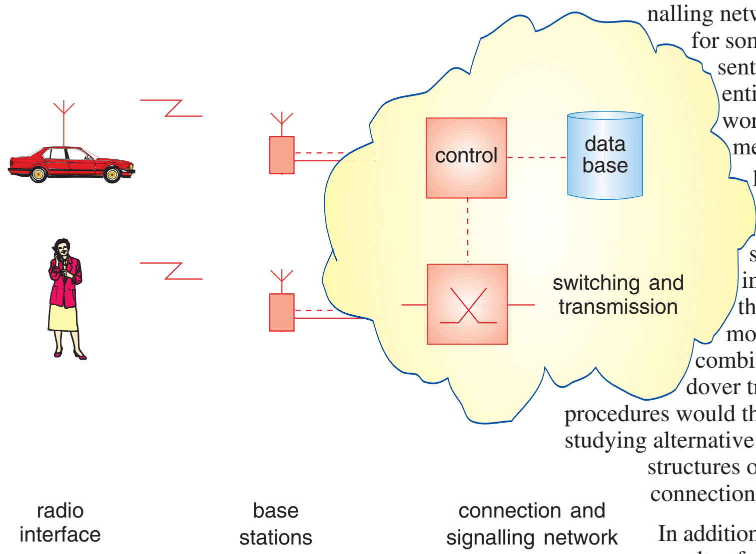


Figure 8 Two view points for estimating the service quality variables: First as seen from the mobile stations, and second, related to the fixed network capabilities

In addition to using the analysis results of such a model when studying time delay variables, the output could be used as input of mobility variables when signalling and processing for the relevant fixed network entities are examined. The teletraffic model can also be seen in relation to a dependability study. For instance, what will be the effects and which measures should be taken when a set of base stations gets reduced traffic handling capability. These effects may not be intuitive in a hierarchical cell structure with overlapping coverage areas.

This discussion shows that the teletraffic performance could be used both as a "stand-alone" result or as a module/step in a wider analysis. Anyway, it is necessary for the model to provide estimates for the relevant variables and allow the analyst to change the input data in order to take into account the aspects studied.

The model must be capable of considering the relevant radio conditions and the service characteristics experienced by the base stations. Using this model, dimensioning of the fixed network part may be performed in order to achieve the stated service goals. For some activity patterns of the mobile stations, we should then be able to estimate the demands for radio resources, processing capacities, signalling links, etc. The influence from changes in the mobile stations' activity patterns on the capacities of the fixed installations can be studied in this way.

4.2.2 Describing the base station coverage

For an outdoor environment, predicting the path loss can be seen as a three step process:

- step 1: predict the link parameters from the terrain data, distance, terrain profile, etc.
- step 2: consider system variables like frequency, antenna heights, etc., and
- step 3: include losses caused by isolated phenomena, like shadowing, etc.

A medium scale prediction model for the total path loss will then result, i.e. aspects like multipath fading are not taken into account. The multipath effect is often related to variations in the signal over very short distances (fraction of wavelengths). It seems difficult to include such variations in a teletraffic model in a direct way. However, as the statistical description of this effect is related to the environment, it could be taken into account in an indirect way.

Several models for radio signal propagation have been described in a number of publications. In order to predict the influence from the various types of base stations to the various environments it could be necessary to look at the cross-effects from the base stations to the other types of environments. For instance, how the signal from a microcell base station propagates through the rooms at the same floor in a building. Anyway, models and measurements related to many of the expected situations are available. This implies that from a given area we could be able to predict the radio signal condition received from the relevant set of base stations.

Radio signal reception is not only given by the static propagation conditions but also influenced by time varying effects like the motion of mobile stations and other interfering, reflecting and screening objects. A moving mobile station will experience fading in particular when Rayleigh propagation models could be appropriate. The duration, level and frequency of fading may depend on the mobile station's velocity relative to the surrounding objects but could also depend on the densities and characteristics of those objects.

Signals from other mobile stations and base stations may interfere with the connection between one mobile station and the corresponding base station. The interference level can depend on the activity level, i.e. the number of connections and the types of connections/services (variable bit rate services). Both co-channel and neighbour channel interference may occur. Shadowing characteristics vary according to both short term and long term time effects.

Most of the effects mentioned in this section could be characterised by some statistical measures. However, it is difficult to consider such phenomena in a general way. That is, we might have to look at each case, and then both the place and the time for the various conditions. As an example, we could use some set of the parameter values for the summer (e.g. because of leaves on the trees) and another set for the winter season.

5 Proposed teletraffic model

The major differences between future wireless systems and the present systems are multiple, like:

- the use of mixed cell structures with overlapping coverage areas of different sizes
- the support of a number of services where each service may use a certain amount of the radio capacity
- the presence of a number of mobile station classes, characterised by their service usage, their velocities, transceiver capabilities, etc.
- the fact that the mobile services will be offered in several environments and by a number of operators/providers.

The first observation, but also the others, invite for the definitions of a new teletraffic model.

5.1 Space element as the area unit

5.1.1 Definition of space elements

In the last two sections some factors for describing the quality of the radio signal have been discussed. The area which we want to model will be named the relevant area. A fundamental idea is to divide this area into a number of space elements. One way of defining a space element is to say that it is the volume having relatively homogeneous quality of the radio signal regarding the relevant base stations, as experienced by the receivers. As returned to later, several additional factors could be included when identifying the set of space elements. The relevant area can then be divided along all three physical dimensions. An example where this is relevant is inside a building with several storeys. Here, some of the base stations may cover different rooms on the same floor and other base stations cover the rooms in the floors above and below the one that is examined.

By relatively homogeneous quality is meant that the signal characteristics may vary within specified limits. In general, both an upper and a lower limit can be given. By proper setting of these limits we are able to model situations ranging from a single area for each base station down to very small-sized elements.

During the division into elements, only the base stations that may serve mobile stations in the area are considered. This can also be influenced by the scope of modelling that has been chosen, e.g. which base stations we want to take into account.

This division into space elements should be performed in the same way as the receivers will experience the signal quality. Various types of receivers may be present and again, we are able to choose to what extent such differences are considered. For instance, we may take into account all subtleties of the receivers and then get a fine structured set of elements (small volumes).

Such a definition allows for a high degree of flexibility when choosing the appropriate level of detail. This concerns the setting of the limits for homogeneous signal quality, the type of receivers taken into account, which radio characteristics (propagation loss, time dispersion, etc.) that are used, and so forth.

5.1.2 Space element identification

As explained in the previous section, we have a flexible model in the sense that the analysts are able to select the level of detail for which the various factors are considered. For coarse models we could simply do this division into space elements based on previous experiences for similar areas. Prediction models can be used to get more accurate values for the relevant space. Such models could further be combined with measurements for the relevant area.

The term relatively homogeneous quality can also be adjusted in line with what we want to study. If several services are implemented and the bit rates are dynamically adjusted according to the signal quality, it might be suitable to set the limits for the signal quality corresponding to the thresholds when the bit rates may be changed. Besides dynamic bit rates, thresholds for the user perception of the service may also be defined. By doing this we can estimate how the mobile users would experience the service.

In principle, this division into space elements can be done in two steps: firstly given by the signal quality from each base station individually, and secondly considering the composite coverage map that results from the first step. This is illustrated in Figure 9 for two base stations. In the figure, two coverage areas are shown for each of the base stations. This could be a result of the fact that it is relevant to consider different levels of the radio signal for the various distances from the base stations. Again, we could neglect the smallest elements if it is not suitable to model that area with a high level of detail. If we model the area with one element for each base station (an element will then be equal to a cell), the intersections of cells may be considered as separate space elements similar to area A in Figure 9. However, the idea of using space elements

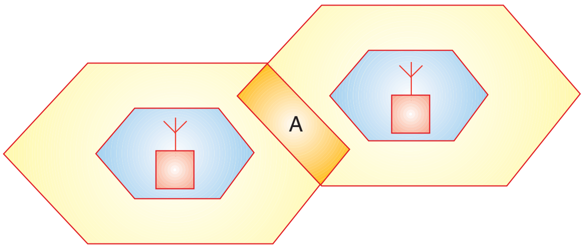


Figure 9 One way of defining space elements: First to identify the coverage area(s) for each base station, and second, to locate the area having similar radio conditions considering the set of available base stations

allows such a model to consider, in addition to situations with non-overlapping cells, more complex coverage patterns. The latter is expected to be the case for systems having several types of base stations. As previously stated, the main reason for introducing the term space element is to allow the analysts to choose the level of detail that will be examined. An additional feature is that this level of detail can vary for the modelled area. For instance, some streets are considered with a high level of detail while others are included on a coarser scale. A reason for this could be to limit the number of variables included in the problem and the amount of relevant input data.

The identification of space elements in a real system will be influenced by the solutions that are selected, like channel coding, multiplexing technique(s) and power classes. That is, the elaboration of guidelines for such identifications should be postponed until more of the system mechanism and features are defined.

So far, we have mainly focused on the coverage patterns when discussing the identification of space elements. Other factors can be included as well. For instance, consider a base station that covers parts of two streets. One street is restricted for vehicles (pedestrians only). Furthermore, assume that only one class of mobile stations is considered. By defining one space element for each of these streets, the input data could be estimated for each of them separately. If only one element was used for this case, we had to estimate the mobile stations' behaviour as seen for both of the streets in combination. This would most likely give other characteristics for the variables as if the first solution was chosen.

5.1.3 Some dynamic aspects

A number of time varying effects will affect the division into space elements, like the weather (e.g. rain) and the level of traffic (interference and noise). In the present state such dynamics are not considered. A simple, yet perhaps demanding solution could be to estimate the boundaries for the new space elements as the conditions are changed and perform the calculation under these changed conditions. Another way could be to say that the space elements may have a range of signal quality levels, each level associated with a certain probability. This would imply that the model (and the analysis) should be enhanced. Therefore, the first way seems to be more inviting. However, in a practical application one set of input data will often suffice for a given configuration. Alternatively, this data could reflect some assumable worst case situation. What will be the worst case may vary depending on the input data that are considered.

5.2 The stochastic processes related to space elements

After defining the space elements we must describe the stochastic processes related to the division of the area into these elements as illustrated in Figure 10. The stochastic processes are mainly the mobile stations' activity patterns, like routing probabilities, dwelling times and call initiating processes. In general, these can be different for every space element in the relevant area.

5.2.1 Classes of mobile stations

A number of mobile terminal types may be present, each type characterised by their service usages, calling intensities, dwelling times and available base stations in addition to other radio variables like the maximum transmitted power, technical solutions for the receiver part (diversity, signal combination, channel equaliser, etc.). Each class that we want to study must be described by the relevant stochastic processes. This allows the analyst to define more classes of mobile stations than commonly used in the standardisation documents related to the mobility categories.

Again, the level of detail is chosen according to the purpose of the analysis. All mobile stations could be handled as one class or a high number of classes could be defined. By using several classes, the characteristics for each may be estimated and given as input in a more specific manner (e.g. with lower variability) than if the combined characteristics of mobile stations with a wide range of behaviour patterns were to be estimated. However, this may depend on the actual situation. The intention of these modelling principles is to allow us to consider the requested phenomena, that is, to allow as much flexibility as possible in correspondence with the expected flexibility for such a mobile system.

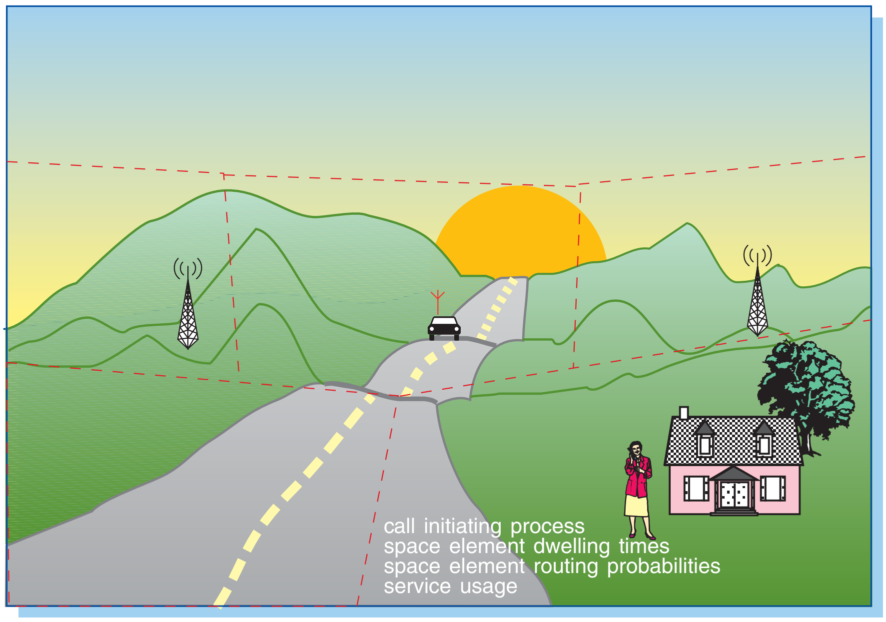


Figure 10 Stochastic processes related to space elements

Although it seems tempting to define a lot of classes, the drawbacks are the dimension of the problem and that the required input data must be given for each class.

5.2.2 Routing probabilities

The mobile stations follow some routes as defined in the set of space elements. That is, given that a mobile station stays in a space element, the probabilities for entering each of the elements as the next one are assumed to be given. Implicitly, it is then said that no historical data of the route that a mobile station has followed are kept (Markov routing). One way of including the historic information is to define a separate class for these mobile stations. There could be classes of mobile stations having predetermined routing and others that follow alternative routes through the relevant area, e.g. the routes could be described by any combination of branches/trees and loops. The routing probabilities can be estimated by measuring or predicting the physical motion of mobile stations (taking into account the number of active stations, i.e. those engaged in calls). This should be done for each class. If there is no indication that active mobile stations follow a route different from the others, the estimate could be based on data for all the mobile stations of that class.

5.2.3 Call initiating processes

Each mobile station class can have its separate description of processes to initiate new calls. These processes can also be different for the various services, and they may vary between the space elements.

To estimate these processes, actual measurements could be done. Another way would be to use a general call initiating process and modify this by considering how the relevant space elements invite the users to make calls for the various services. Making consistent measurements of the call initiating process may be difficult. In most models of cellular systems new calls are assumed to be initiated according to a Poisson process for each cell. How well such an assumption is for a fraction of a cell should be validated.

5.2.4 Space element dwelling times

The time a mobile station stays in a space element is named the dwelling time for that element. The distribution of the dwelling times may be estimated by predictions and measurements. Here, we may have to consider the possibility that a mobile station had stayed in an element for a while before the call was initiated. That is, there may be a difference in the remaining dwelling time for a mobile station after it has initiated a call in that space element and the dwelling time if it already had an ongoing call when the space element was entered.

These distributions will depend on how the mobile stations move. For an element that contains many mobile stations and where each of these have an independent behaviour regarding the arrival, departure, number of turns and routing, the dwelling times could perhaps be approximated by a negative exponential distribution, ref. [6]. For elements covering a highway with heavy vehicular traffic, an Erlang-k distribution or a truncated Gaussian distribution might be more appropriate as assumptions of the dwelling times.

Dwelling times may differ for the various classes of mobile stations. In fact, the velocities of the mobile stations may be one for describing the factors for dividing these into classes.

5.2.5 Coverage

The base stations' coverage for space elements are assumed to be known by predictions or measurements. Each class of mobile stations may have its separate set of base stations. This set is based on which base stations the mobile stations of that class are allowed to use (authorised). Other reasons could be the available power, signalling processing used, supported services, etc.

As seen from a space element, a mobile station of a given class will then have a set of base stations available. These base stations are assumed to be sorted in a list, called a coverage list. A number of criteria for sorting the base stations could be applied. All classes of mobile stations do not have to use the same set of criteria; neither must they arrange the criteria in the same way. For example, one criterion could be that the emitted power should be as low as possible, resulting in selection of a small-cell base station (pico of microcell) before a macrocell base station. For another class of mobile stations the emitted power may not be a main constraint and we could rather wish to be connected to the same base station as long as possible (to limit the number of handovers) which could result in choosing a macrocell base station before a microcell base stations. The operators would naturally have an influence on how such lists are arranged, for instance, by applying appropriate charging policies.

A natural order would be to let most mobile stations look for the small-cell base stations first. This would often lead to an increase of the traffic handling capacity (capacity reuse) as lower power would be emitted. In addition, the quality of the received radio signal could be higher. If the first base station in a list does not have sufficient free capacity to serve the call, the mobile station should look at the next one in the coverage list. The information necessary for mobile stations to select the "best" possible base station could be sent in a broadcast channel from each base station. Then, the mobile stations must scan these broadcast channels for (some of) the relevant base stations in order to make a decision of which one to use.

5.2.6 Service usage

Mobile stations from a given class may use a set of services that differ from the other classes. On the other hand we could define one class for every service that is used. Each class can also use the services in its own way, i.e. defined by different values and distributions compared to the other classes.

The service usage may be dependent on the space elements, that is, in which environment a call is made. For instance, it seems natural that a telephone call made when standing in a crowd is shorter than if the same call were made sitting in a comfortable chair. On a larger scale, city areas and rural areas might have different characteristic values.

5.3 Characterising the services

5.3.1 Holding times

It must be possible to define different total holding times for the various services. These holding times may also vary between the classes of mobile stations and depend on the environment in which the call is initiated. One factor that might influence this is how the mobile user experiences his/her surroundings (non-human users are also included). The holding times can be measured for longer calls, like telephone calls, and predicted for shorter calls, like short message services when the message lengths are given. Whether it is appropriate to allow for such services to compete for the same radio capacity is not further dealt with here. However, the potential economics of scope for the services was indicated many years ago, see e.g. [11].

5.3.2 Radio capacity

The various services may not use the same capacity on the radio section. A video connection may for instance need a capacity equal to a number of speech connections. The needed capacity may also vary between space elements depending on the radio signal quality even if the same base station is used. That is, if the coding and the multiplexing are dynamically adjusted to take such effects into account. If the used radio capacity is the same, it may not have any impact on the resources used but can affect the service quality experienced by the user.

The used capacity on the radio link will also depend on the multiplexing techniques, coding scheme, discontinuous transmission/voice activity detection, etc. Other factors like one way or two way connections and direct links between mobile stations will also influence how much radio capacity a service needs. The influence can be found by examining the chosen solution, and then combine this by statistics/measurements on how often the different mechanisms will come into effect.

Several access techniques (and variants thereof) have been proposed for a future wireless system. By using the term resource unit, we want to quantify the capacities of the radio interface that calls need without saying how this is done. Therefore, talking about resource units may be seen as an abstraction of the real resources in the radio interface, see Figure 11. When a call has a fixed amount of the capacity assigned, a resource unit could be a slot in a TDMA system or a certain value of the bandwidth power product in a CDMA system [2]. Furthermore, some resemblance with the term logical channel could be recognised.

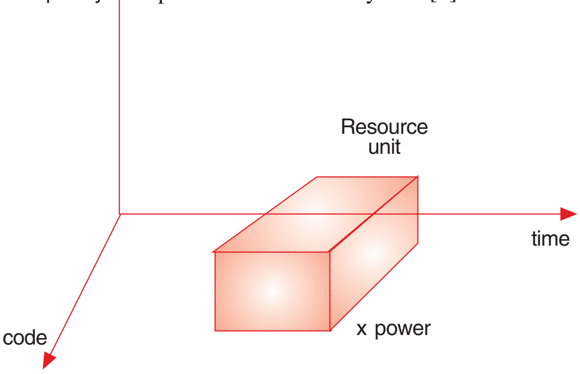


Figure 11 Resource unit – a possible abstraction of the real resources on the radio interface

A definition of the resource units may also be needed when a call admission function is to decide whether the call should be established or rejected. Then, we measure the capacity in the number of resource units. For instance, a call of a given service will require a number of resource units. The hybrid capacity allocation could deal with these resource units as well. By defining them as discrete and countable variables, it is simpler to describe mechanisms and procedures that handle these aspects.

5.4 Network of base stations

Each base station will be modelled as a node in an open queuing network. The queuing network is open as new calls may be initiated and calls may be released. We may also have the possibility for established calls belonging to mobile stations that enter and leave the modelled area.

A connection between two nodes will then correspond to a possibility of handovers between the relevant pair of base stations. That is, the flows between nodes in the queuing network reflect the handovers.

Each base station has allocated a certain amount of radio capacity according to the applied allocation scheme.

5.5 Time varying phenomena

Several of the factors discussed so far in this chapter will not be stationary. Examples of some non-stationary phenomena are:

- activity patterns of mobile stations both as described in the set of space elements and with regard to call activities (new calls and holding times also depend on time of day, day of week, etc.)
- interference and noise will depend on the number of calls, the number of noise sources, etc.
- allocation of radio capacity to base stations for non-fixed schemes.

Another time-varying factor could be the mobile cells, i.e. where the base stations are moving. If these are not allocated separate resources, e.g. a separate frequency band, interference can occur. How severe this interference is also depends on the transmitted power and the screening.

Some other non-stationary phenomena were discussed in Section 4.2. In principle, we could repeat the calculations for the various sets of input data in order to see the impact on the results.

Transient analyses would also be of interest. For instance, to increase the traffic load in a space element to see how long time it would take in order to reach the next (close to-) stationary state. This is especially relevant for non-fixed capacity allocation schemes.

6 Analysing the model

In the model proposed we may say that there are two types of flows: First the flows described in the set of space element by active mobile stations, like call initiation processes, space element dwelling times and space element routing. Second, the flows described in the set of base stations by calls, like call arrival processes (newly initiated calls and handover calls) and serving times. Naturally, these two types of flows are related and the relationships must be found in order to estimate the service quality variables.

The main input data provide estimates for the mobile stations' behaviour patterns and the base stations' coverage. The task of the analysis can be illustrated as in Figure 12.

In order to find the service quality variables in an analytical way, we must establish the "rules" for transforming the flows described in the set of space elements to corresponding flows described in the set of base stations.

As we are considering active mobile stations (i.e. those attempting to make calls or engaged in calls), phenomena in the base station network will influence the flows in the space elements. That is, an analysis procedure based on iteration is usually required. A flowchart for the analysis procedure is depicted in Figure 13. As shown in the figure, the approach can be divided into two parts: In the first part the basic relationships between the flows in the space elements and the base stations are calculated such as the arrival processes to base stations and the base stations' serving times. In the second part the processes in the base station network are looked into in more detail, such as the overflows and the departures. Naturally, these parts are interdependent, and we should continue the iterations until some convergence criteria are met.

As stated above, the base station network is treated like an open queuing network. This is then analysed by parametric decomposition which is an approximate technique, e.g. see [8]. Furthermore, several other approximations are used when calculating the characteristics of the flows in the queuing network.

In order to perform the analysis we must elaborate relationships between the arrival processes/serving processes and the overflow processes, the departure processes and the blocking probabilities. By looking at the problem in this way, a modular procedure is identified. That is, each of the relationships could be improved more or less independently of the others. In addition, we may introduce other features that could be present in a wireless system. Such features include the rearranging of established calls between base stations, allowing calls to wait if a base station does not have sufficient free capacity and hybrid capacity allocation.

There are several variants of the rearranging mechanism, mainly related to the time when an established call can be rearranged and between which base stations connections can be rearranged. Without considering any requirements for the minimum handling time of a connection by a base station, we may say that it is possible to rearrange a connection at any time. That is, as soon as a better base station has sufficient free capacity. An alternative is to say that rearrangements are only possible for connections related to mobile stations when these change space elements, i.e. at boundary crossings. Which base stations a connection may be rearranged to can be given as a sorted list for each base station in the coverage list for the space element. Introducing the rearranging possibility should increase the quality of the connection and/or the overall system capacity. However, the handover rates between base stations will increase.

Allowing calls to queue if a base station does not have sufficient free capacity at the time of arrival should also increase the system capacity. By restricting this feature to some call types, these would get a higher priority compared to the other call types. A time-out mechanism would usually be present in order to limit the time a call could stay in such a queue. For a handover call such a time-out may be given by the time the mobile station is able to continue to use its current base station. That is, the time until the radio signal quality is below a certain limit.

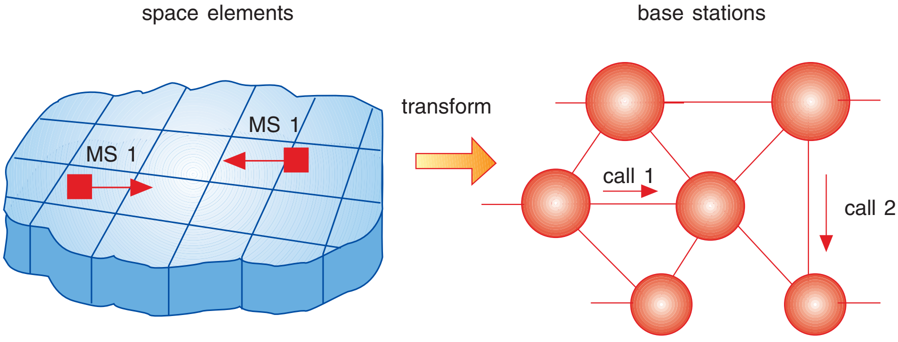


Figure 12 Transforming the flow patterns described in space element to flow patterns described in the base station network

Also for this queuing feature several variants can be defined, e.g. described by which calls are allowed to queue, maximal times in queue, which base stations have the queuing possibility, number of queues in which a call may be placed simultaneously, etc. By varying these factors we are able to identify combinations that would improve some measure for a system implementation in certain situations. As expected, such a queuing possibility has the highest effect when there is a high probability that a call under service will leave the base station before the maximum queuing time elapses.

The study of algorithms for allocating capacity to base stations is a field of much interest. A number of relevant criteria and procedures for how to allocate the capacity (when to allocate capacity, which capacity to allocate, etc.) can be defined. If a variant of hybrid schemes is applied, we must also describe the algorithms for when some capacity can be returned from a base station to a common pool.

For a multi-operator, mixed cell structure configuration we must define the ways to share the capacity between the different operators and cell types. This could also be influenced by the access and other radio link techniques that are chosen.

In addition to analytical calculations, simulations can be carried out. An example of this is given in the following chapter. Naturally, simulations would also be performed for evaluating the accuracy of the results found from calculations.

7 An example

The application of the model will be illustrated through an example. The examined structure is schematically depicted in Figure 14. There are 12 space elements identified from [1,1,1] to

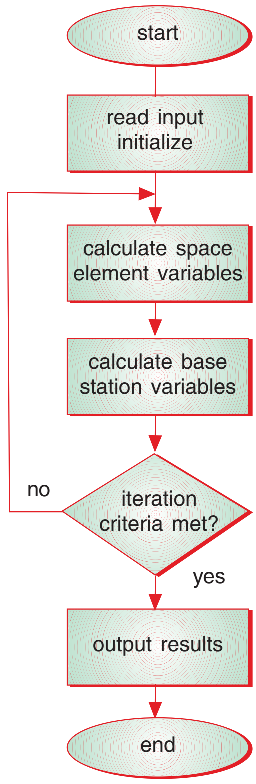


Figure 13 Simplified flowchart for the analysis algorithm

[5,2,1]. Three classes of mobile stations have been defined with a routing as shown in the figure. The routing is "wrapped" with the meaning that mobile stations leaving element [5,1,1] enter [1,1,1], and so forth. Six base stations are used in the basic configuration. The base stations are arranged into two layers. The coverage list for each of the space elements is given in Table 1. A mobile station initiating a call or performing a handover will start looking for free capacity in the first base station in the relevant list. All the space elements in a column have the same coverage list.

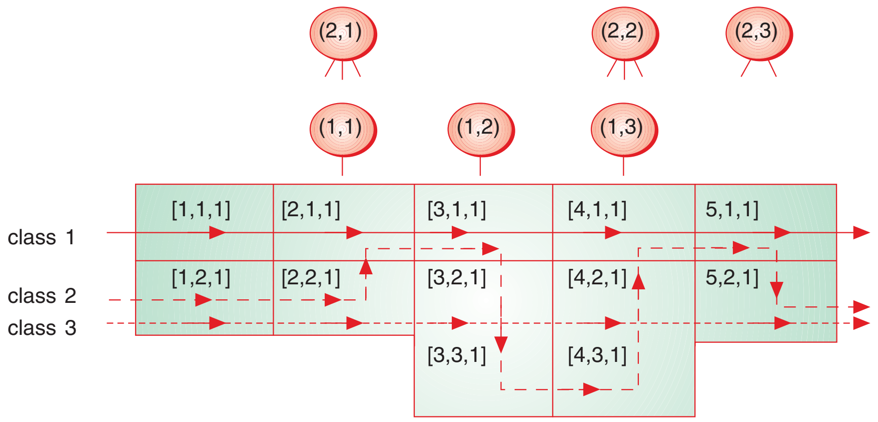


Figure 14 Schematic illustration of the example: 12 space elements, 3 classes of base stations and 6 base stations (arranged in two layers). The space element routing is "wrapped"

Base stations in layer 1 have 8 resource units allocated to each of them. 12 resource units have been allocated to each of the base stations in layer 2.

Some data for the mobile station classes are given in Table 2. Both the space element dwelling times and the total call holding times are negative exponentially distributed. Every space element has the same mean dwelling time. The process of initiating new calls is equal for all the classes and in each of the space elements. New calls are initiated according to a Poisson process and the rate is used as the free variable in the results given.

Five different cases of the configuration have been compared. These cases are defined in Table 3. Case I is the one presented so far. This is used as reference for the other cases, i.e. only modifications compared to case I are described in the following. In case II the class 1 mobile stations are allowed to use base stations in both layers. An additional base station (3,1) is introduced in case III. This base station covers all the space element and is reserved for mobile stations of class 3. Then, these mobile stations are not allowed to use base stations in layer 2. Base station (3,1) is capable of handling three mobile stations of class 3 simultaneously (6 resource units allocated). The number of resource units in the layer 2 base stations is reduced by 2 each (that is, 10 in each of layer 2 base stations). In case IV a hybrid capacity allocation scheme is introduced. Each of the base stations is allocated 4 resource units as fixed. A maximum amount of resource units possible to allocate a base station has been set. A layer 1 base station cannot have more than 12 resource units and a layer 2 base station cannot have more than 20 resource units. The capacity is allocated in groups of 4 resource units. There are 9 groups of resource units that can be allocated dynamically. An interference matrix is defined. This matrix states that a resource unit cannot be allocated to more than one base station at a time (that is, one base station interferes with all the others). Queuing of handover calls are allowed for in case V. Every base station has a queuing capacity of 2 calls. The allowed time in a queue is negative exponentially distributed with a mean value of 0.1, a maximum queuing time of 0.2 is set. A handover call is allowed to wait in the queue for up to 5 base stations at the same time (this is a non-effective limit as no space element is covered by so many base stations).

The space element blocking probability is considered to be the main service quality variable for this example. The blocking probabilities that mobile stations of class 1 experience in space element [4,1,1] are depicted in Figure 15. We see that the different cases influence the blocking probability. The most significant effect results for case V (queuing of handover calls is allowed). In this case, the blocking of handover calls is the resulting effect of the queue being full upon arrival and that the time-out interval ends before the call can be served by any of the base stations it has placed in the queue. As seen from the mobile station, the effect of each of these events will be similar. For case V, the new class 1 calls get reduced blocking while the handover calls of class 1 get an increased blocking. An explanation to the latter is that some class 1 handover calls may face a situation where a class 3 handover call is located in front of them in the queue. As this class 3 call requires 2 resource units, it is more likely that it stays in the queue until the time-out interval is ended. Therefore, the class 1 handover calls are experiencing a higher blocking, which again would allow for more new calls to be handled by the base stations. We also see that introducing hybrid allocation of capacity, case IV, does not seem to improve the situation much in this example compared to case I. Allowing the class 1 calls to use base stations in both layers, case II, gives lower blocking for this class. This was expected. In case III, the class 3 calls are not competing with class 1 and 2 calls for capacity of base stations in layer 2. For the results depicted in Figure 15, we see that this results in lower blocking for class 1 calls for lower load, when compared to case III. However, for higher load the result is the opposite. An explanation of the latter may be that each of the base stations in layer 2 has got reduced capacity in case III compared to case II and this effect is more significant when the traffic load is high.

Table 1 Coverage list for the space elements in Figure 14

Space elements	Coverage list
[1,1,1], [1,2,1]	{(2,1)}
[2,1,1], [2,2,1], [2,3,1]	{(1,1), (2,1)}
[3,1,1], [3,2,1], [3,3,1]	{(1,2), (2,1), (2,2)}
[4,1,1], [4,2,1], [4,3,1]	{(1,3), (2,2), (2,3)}
[5,1,1], [5,2,1]	{(2,2), (2,3)}

Table 2 Relevant times and radio capacity usage for mobile station classes

mobile station class	mean space element dwelling time	mean total call holding time	available base stations (layers)	number of resource unit per call
1	1	8	2	1
2	5	8	1 and 2	1
3	2	8	1 and 2	2

The space element blocking probabilities for class 2 calls in element [4,2,1] are shown in Figure 16. We see that allowing class 1 calls to use base stations in layer 1 (case II) has a minor impact on the blocking of class 2 in this example. Although it is not very clear in the figure, the blocking in case II is always higher than the blocking in case I. The other cases give a reduced blocking for class 2. Case III and case IV show more or less the same blocking in this example. Introducing hybrid capacity allocation (case IV) results in lower blocking for situations with low load when compared to case I. This effect is clearer for class 2 than for class 1. In case V, the time-out effect for handover calls is expected to give significantly lower blocking for new calls as explained for class 1. There is no need for handover when a class 2 mobile station with an ongoing call enters space elements [4,2,1].

Class 3 space element blocking probabilities in [4,2,1] are depicted in Figure 17. As for class 2, we see that class 3 calls get higher blocking when class 1 calls can use base stations in both layers (case II), compared to case I. Apart from this, the different cases for class 3 are more or less the reverse of class 2. That is, case III, IV and V give higher blocking than case I. This may be an effect of the fact that improving the condition for one class can make it worse for another class; a trade-off may emanate. Naturally, this is an effect of a class 3 call requiring 2 resource units while calls of the other classes require 1 resource unit each. Although the resulting blocking in case V is higher, we see the effect of allowing handover calls to be queued. That is, the handover calls have a lower blocking compared to new calls in this example.

The results shown in Figures 15, 16 and 17 are only indications of which effects the different configurations and features can have. These are all referring to two of the space elements in the examined example. If results from other space elements were depicted, the different cases could come out in another order. Therefore, the observations stated above can not be generalised without further examinations. Moreover, the results will depend on the load, the mobility, the service usage and the configuration.

In addition to the space element blocking probabilities, several other results could have been presented; like the traffic load and individual blocking probabilities for each of the base stations.

Table 3 Description of the cases

case I	As explained above; used as reference.
case II	Class 1 mobile stations are allowed to use base stations in both layers.
case III	Additional base station (3,1) with 6 resource units, reserved for class 3, covers all the space elements. Class 3 is not allowed to use base stations in layer 2. Layer 2 base stations have 10 resource units each.
case IV	Hybrid allocation. Fixed part of 4 resource units to every base station. Maximum limit of 12 and 20 resource units to a base station in layer 1 and layer 2, respectively. Capacity is allocated in groups of 4 resource units. 9 groups in the pool can be allocated dynamically.
case V	Queuing of handover calls. 2 queuing positions in each base station. Allowed queuing time is ned with mean 0.1 and a maximum value of 0.2. A handover call can be placed in the queue of up to 5 base stations simultaneously.

8 Conclusions

The request for mobile services seems to be steadily increasing. Even though digital systems have been introduced, it is expected that the traffic demand for such services will require new systems soon after the turn of the century.

The definition of such future wireless systems is carried out by several organisations. As the well used phrase “any one, any time, any place, any form” goes, these systems could be characterised by a multitude of operation environments, operators, services and mobile station classes.

This observation requests the elaboration of teletraffic models to describe the situations expected to arise. As present systems can be seen as subsets of such future systems, the teletraffic model will be applicable for those as well. Such a model and corresponding analysis have been outlined. Naturally, as several mechanisms remain to be described for future wireless systems, it is difficult to calculate the service quality variables with any confidence for a real case. However, the model/analysis can be used to examine the effect of possible features that could be introduced in such systems. The example indicates that the effects of varying the configuration and introducing features may not be easy to intuitively determine in advance. Therefore, after estimating the requested input data, different configurations and features could be examined by applying the model and analysis.

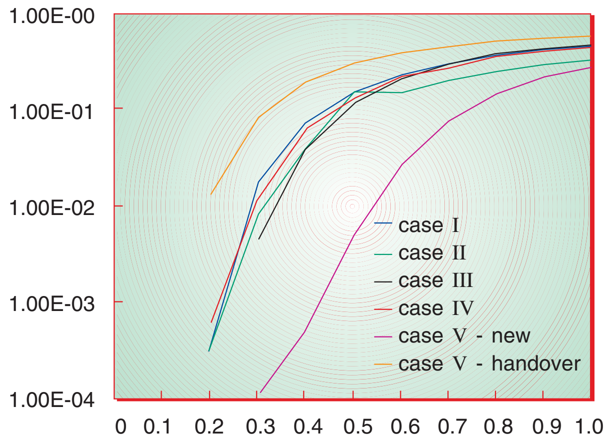


Figure 15 Space element blocking probabilities for mobile stations of class 1 in element [4,1,1]. The different cases are indicated. For case V, new calls and handover calls experience different blocking

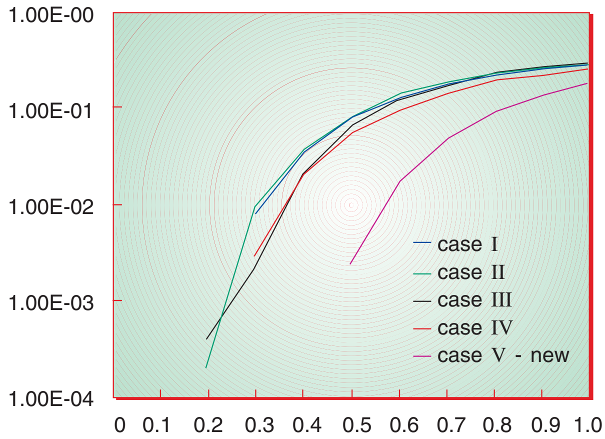


Figure 16 Space element blocking probabilities for mobile stations of class 2 in element [4,2,1]. The different cases are indicated. There is no need for handover when class 2 mobile stations with an ongoing call enter element [4,2,1]

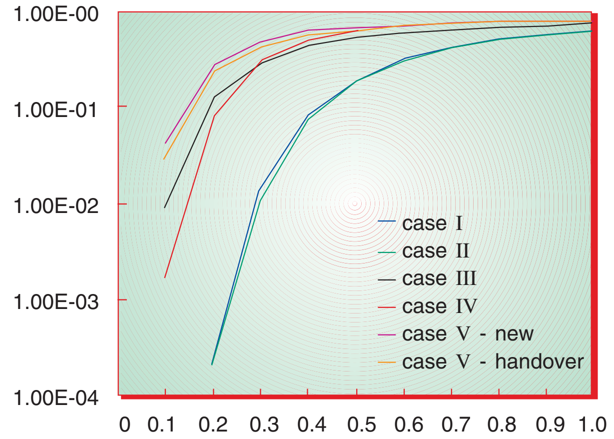


Figure 17 Space element blocking probabilities for mobile stations of class 3 in element [4,2,1]. The different cases are indicated. In case V, new calls and handover calls will experience different blocking/time-out as queuing is allowed

Abbreviations

ADC	American Digital Communications system
AMPS	Advanced Mobile Phone System
CDMA	Code Division Multiple Access
CT	Cordless Telephone
DECT	Digital European Cordless Telecommunications
ETE	Equivalent Telephone Erlang
ETSI	European Telecommunications Standards Institute
FDMA	Frequency Division Multiple Access
FPLMTS	Future Public Land Mobile Telecommunication Systems
GSM	Global System for Mobile communications
HIPERLAN	High Performance LAN
IN	Intelligent Network
ITU	International Telecommunication Union
JDC	Japan Digital Communications system
LAN	Local Area Network
MAN	Metropolitan Area Network
MT	Mobile Termination
ned	negative exponentially distributed
NMT	Nordic Mobile Telephone
PABX	Private Automatic Branch Exchange
PCS	Personal Communication System
SIM	Subscriber Identity Module
SMG	Special Mobile Group
TACS	Total Access Communications System
TDMA	Time Division Multiple Access
TGMS	Third Generation Mobile Systems
UMTS	Universal Mobile Telecommunications System

References

1 Andermo, P G, Larsson, G. Code Division Testbed, CODIT. I: *2nd international conference on universal personal communications,* Ottawa, Canada. 12 – 15 Oct. 1993, 397–401.

2 Baier, A. Multi-Rate DS-CDMA : a promising access technique for third-generation mobile radio systems. I: *4th international symposium on personal, indoor and mobile radio communications,* Yokohama, Japan. 8 – 11 Sep. 1993, 114–118.

3 Calhoun, G. *Digital cellular radio.* Artech House, USA, 1988.

4 Grillo, D et al. A simulation testbed for the analysis of short-term adaption mechanisms in the ATDMA system. I: *Mobile and personal communications : IEE conference publication,* 387, 1993, 216–220.

5 Goodman, D J. Trends in cellular and cordless communications. *IEEE communication magazine,* June 1991, 31–40.

6 Guèrin, R A. Channel occupancy time distribution in a cellular radio system. *IEEE transactions on vehicular techn.,* VT-35, 3, 1987, 8999.

7 Jensen, T, Myskja, A, Larsen, E. Performance of dynamic capacity allocation in future wireless communication systems. *Wireless computer networks (WCN-94).* The Hague, The Netherlands, 21 – 23 Sep. 1994, 1014–1018.

8 Kuehn. P J. Approximate analysis of general queuing networks by decomposition. *IEEE transactions on communications.* COM-27, 1, 1979, 113–126.

9 Nanda, S, Dravida, S, Samadi, B. Handoff management and performance for wireless access using metropolitan area networks. I: *8th ITC Specialist Seminar on UPT,* Santa Margherita Ligure, Italy, 1992, 191–199.

10 Nanda, S. Teletraffic models for urban and suburban microcells: cell sizes and handoff rates. I: *8th ITC Specialist Seminar on UPT,* Santa Margherita Ligure, Italy, 1992, 251–260.

11 Peritsky, M M. Traffic engineering of combined mobile telephone and dispatch systems. *IEEE transactions on vehicular techn.,* VT-22, 4. Nov. 1973, 223–225.

12 Samadi, B, Wong, W S. Optimisation techniques for location area partitioning. I: *8th ITC Specialist Seminar on UPT,* Santa Margherita Ligure, Italy, 12 – 14 Oct. 1992, 141–147.

Analysis of external traffic at UNIT/SINTEF's MD110 telephone exchange

BY BONING FENG AND ARNE MYSKJA

1 Introduction

Since the 1950s three generations of automatic telephone exchanges have been installed at the University of Trondheim (UNIT):

1 A main exchange of type AGF (delivered by Ericsson), electromechanical type, together with 17 smaller exchanges geographically distributed both in and outside the main campus area. The exchanges covered UNIT and the research institute SINTEF, and was in service until May 1977.

2 A Metaconta 10R exchange (delivered by LMT Paris), a computerised system with mechanical contacts. It covered UNIT and SINTEF except some remote units. From 1982 the Regional Hospital of Trondheim (RiT) was also connected to the same exchange. This exchange was in service from May 1977 to January 1990.

3 In January 1990 a new telephone exchange, MD110 (delivered by Ericsson) was installed at UNIT/SINTEF. The exchange is fully electronic, digital, and distributed. It consists of 36 switching units (LIMs) on 16 different sites which are connected by a star network with altogether 54 internal 2 Mb/s leased circuits. The new exchange covers all the units of UNIT and SINTEF, but not RiT.

This paper presents some results from the traffic analysis at the new exchange. Due to the limitations of the registration functions implemented in the MD110 it is not possible to make detailed studies of the incoming traffic. Another type of registration mechanism which MD110 lacks, is the possibility to register the distribution of the internal traffic. Therefore our study is focused on direction-based analysis of outgoing external traffic at the telephone exchange. The results presented here cover

- number of calls
- traffic volume, and
- duration of calls.

The study includes traffic profiles of 24 hour and full week periods, and how they are distributed among the different service types and charging groups. We also look at the number of calls and the traffic to the other universities in Norway. The number of incoming call attempts was also registered in order to get an indication of the relation between incoming and outgoing calls.

In addition to averages, distributions are also included. Histograms are given for some distributions. In most cases mean, standard deviation and coefficient of variation are calculated.

2 Traffic observations

2.1 Overview of the telephone exchange at UNIT/SINTEF

At the time of observation the system consisted of the following parts:

- 36 LIMs
- 54 leased circuits, each with thirty 64 kb/s channels
- 2 group selectors in load-sharing mode
- 7 two-way ISDN primary access with a total of 210 channels
- 2 outgoing PCM groups with a total of 60 channels
- 40 outgoing analogue reserve lines
- 6500 extensions.

2.2 The data registration system

Traffic observations have been undertaken on all three generations of telephone exchanges:

- on AGF exchange by analysis of the charging data stored on magnetic tapes, collected by a charging system developed by NTH (The Norwegian Institute of Technology) [1]
- on Metaconta 10R by analysis of the automatic registrations made by the system, and by corresponding analysis of the charging data [2, 3, 4]
- on MD110 by data registered by the supporting system TELMAX, and by analysis of the charging data [6].

Despite the limitations mentioned above, the traffic registration in MD110 is still quite extensive, and they are made available through the supporting system TELMAX. Combined with the charging data generated by TELMAX, the data sets for the following types of analysis are available:

- number of call attempts, traffic intensity and volume at all server groups, internal and external trunk groups
- number and holding times of calls, traffic intensity and volume, distributed on arbitrary external traffic directions, defined by the numbers called (direction-based analysis).

The direction-based analysis is based on the charging data. In order to secure the privacy of the employees, all the calling numbers have been deleted during our analysis. Neither is any analysis based on particular numbers.

The supporting system TELMAX is also distributed, using the broadband network already installed in UNIT/SINTEF and local Ethernet-segments. The system is PC-based (at UNIT/SINTEF IBM PS2 are used). There are altogether 21 PCs in the TELMAX system.

The traffic module in TELMAX generates directly graphical curves showing the number of call attempts and the traffic. The output data are averages, i.e. measurements of load on server groups like leased circuits, signalling equipment, etc.

For incoming traffic there is no direct registration mechanism. However, it is possible to measure

- number of call attempts
- number of calls which are set up manually (through the exchange switchboard)
- total traffic (sum of incoming- and outgoing traffic).

These registrations are not done automatically; they must be initiated manually.

The information of outgoing calls are given by the charging data, which supply detailed information of every single successful call, including B-numbers. A typical record in a charging file is shown in Figure 1. It is possible to make analysis as detailed as desired. This kind of analysis has been performed earlier for the purpose of studying the statistics for repeated calls [1]. In the present study all the A-numbers have been deleted, so the direction-based analysis gives collective results for the whole exchange.

2.3 Arrangement of the observations

2.3.1 The observation periods

The charging files from two observation periods in the 4th quarter of 1991 have been collected:

- Period 1: Monday 1991.10.07 – Sunday 1991.10.20
- Period 2: Monday 1991.11.25 – Sunday 1991.12.08

Both periods have a two week duration. Since October is considered a "normal

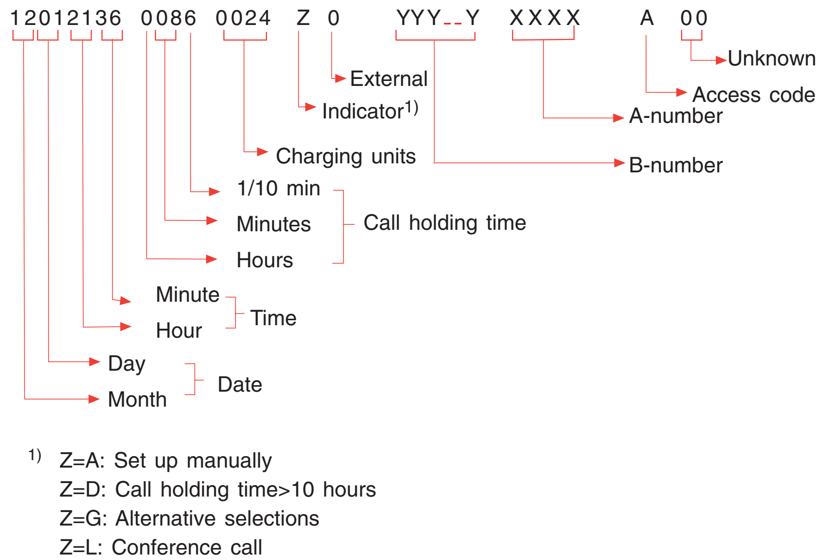


Figure 1 A typical record in a charging file

period” (i.e. not close to vacations) we decided to analyse data from Period 1 in more detail. The results from Period 2 are used to check the results from Period 1.

By using TELMAX some additional information is obtained. The measurements were taken in the following periods:

- Period 3: Monday 1991.11.25 – Friday 1991.11.29
- Period 4: Monday 1992.01.13 – Tuesday 1992.01.14

During Period 3 (which coincides with the 5 first weekdays of Period 2) the number of incoming and outgoing call attempts, and the sum of the incoming and outgoing traffic were measured between 0900 and 1900. Measurements during Period 4 were made in order to get the percentage of incoming calls using the manual service of the switchboard.

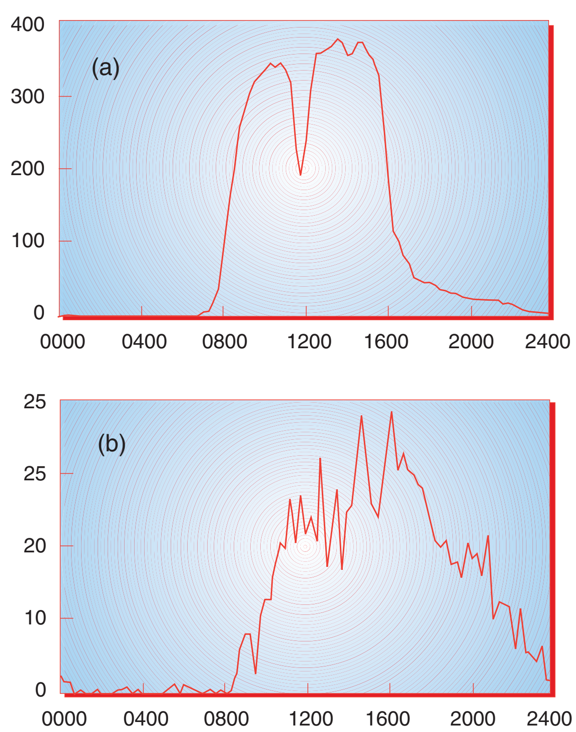


Figure 2 Mean number of successful outgoing calls per 15 minute intervals. (a) weekdays, (b) weekends

2.3.2 The address groups

In addition to analysis of the total traffic we also investigated the distribution of traffic between different address groups. The constitution of address groups is based on service types and charging groups. We also observed the traffic to the other universities of Norway (Universities of Oslo, Bergen and Tromsø). Another interesting task is to find the percentage of all traffic which goes to Oslo and its neighbouring counties.

The following address groups have been defined:

- local calls (within the county limit)
- calls to the neighbouring counties (of Trondheim)
- domestic long-distance calls
- international calls
- paging
- calls to mobile telephones
- special services numbers (e.g. customer services provided by Telenor AS, emergency calls, etc.)
- toll-free numbers,

and, as we mentioned before,

- The University of Oslo (UiO), The University of Bergen (UiB) and The University of Tromsø (UiTø)
- Oslo and its neighbouring counties,

which all belong to the address group “domestic long-distance calls”.

By using the telephone directories from Telenor and the universities we are able to map the numbers belonging to the different categories.

A view of which combinations are analysed in this study is shown in Table 1.

3 Results and discussions

3.1 Profile of 24-hour periods

All successful outgoing calls are registered in the charging files. The number of calls per 15-minute intervals have been summarised for the two-week period. Figure 2 shows the time diagrams for the average number of calls per interval, for weekdays and weekends, respectively. (Note different ordinate scales.)

The diagrams show that the call intensity is negligible between midnight and 0700. For weekdays the call intensity increases until it reaches a small peak around 1030. After a clear reduction during the lunch period between 1100 and 1230, it increases again to a higher and longer peak which continues until after 1500. After that, the call intensity decreases sharply until 1700, and continues decreasing smoothly until midnight.

The observation on the traffic volume shows the same tendency as that of number of calls.

Some earlier traffic studies of public telephone exchanges [5] show a high peak before the lunch period (between 0900 and 1000) and no distinct peak was observed after the lunch period. Our results show a different traffic pattern than normal public telephone exchanges. We agree with the explanation given in [4] saying that many of the users of the

UNIT/SINTEF exchange probably choose to do the easier routine work (mostly administration work which usually requires more use of the phone) in the last part of the day. We also believe that many professors (who constitute a large part of the users connected to the exchange) are more involved with lecturing during the first part of the day.

As expected there are much fewer telephone calls during weekends. The 24-hour profile shows a similar tendency, but the peak occurs later and the profile is less influenced by normal working hours. The calls are not as concentrated and regularly distributed as on weekdays.

Figure 2 does not show the differences between address groups. In order to register this we make some more analysis by dividing all outgoing calls into 3 groups:

Table 1 Overview of the combinations studied
a Group 3 covers traffic to the address groups 1, 2, 7, and 8 ("inexpensive calls")
b Group 2 covers traffic to the address groups 3, 5, and 6 ("more expensive calls")

Outgoing traffic	**Number of calls**	**Traffic (Erlang)**	**Call holding time**
Total	24-hour periods and week profile mean and st.dev. per weekday 24-hour periods profile of no. call attempts no. manually set-up calls	24-hour periods and week profile busy hour (for each weekday) time consistent busy hour	mean, st.dev. and histograms
1) Local (within Trondheim)	mean and st.dev. per weekday	mean and st.dev. 24-hour periods profile (gr.3[a])	mean, st.dev. and histograms
2) Neighbouring counties	mean and st.dev. per weekday	mean and st.dev. 24-hour periods profile (gr.3)	mean, st.dev. and histograms
3) Domestic long-distance			
Total	mean and st.dev. per weekday	mean and st.dev. 24-hour periods profile (gr. 2[b])	mean, st.dev. and histograms
3a) Universities	mean and st.dev. per weekday (for each university)	mean and st.dev. (for each university)	mean, st.dev. and histograms
Universities, fax	mean and st.dev. per weekday (for each university)	mean and st.dev. (for each university)	mean, st.dev. and histograms
3b) Oslo	-	mean	-
3c) Oslo, neighb. counties	-	mean	-
4) International calls	mean and st.dev. per weekday	mean and st.dev. 24-hour periods profile	mean, st.dev. and histograms
5) Mobile telephone	mean and st.dev. per weekday	mean and st.dev. 24-hour periods profile (gr. 2)	mean, st.dev. and histograms
6) Paging	meand and st.dev. per weekday	mean and st.dev. 24-hour periods profile (gr. 2)	mean, st.dev. and histograms
7) Special services	mean and st.dev. per weekday	mean and st.dev. 24-hour periods profile (gr. 3)	mean, st.dev. and histograms
8) Toll-free numbers	mean and st.dev. per weekday	mean and st.dev. 24-hour periods profile (gr. 3)	mean, st.dev. and histograms
Incoming traffic			
Total	24-hour periods profile of no. call attempts no. manual set-up call attempts	24-hour periods profile	-

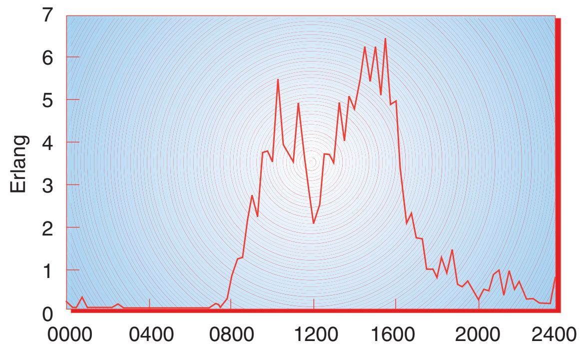


Figure 3 Outgoing traffic (average of weekdays during Period 1), Group 1

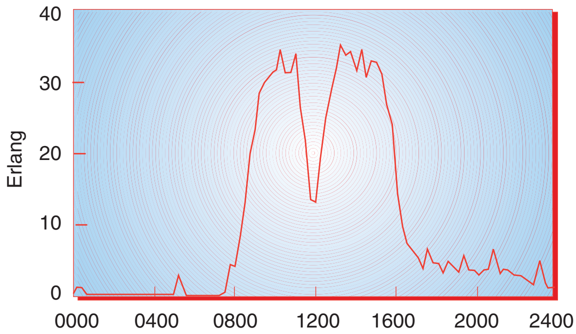


Figure 4 Outgoing traffic (average of weekdays during Period 1), Group 2

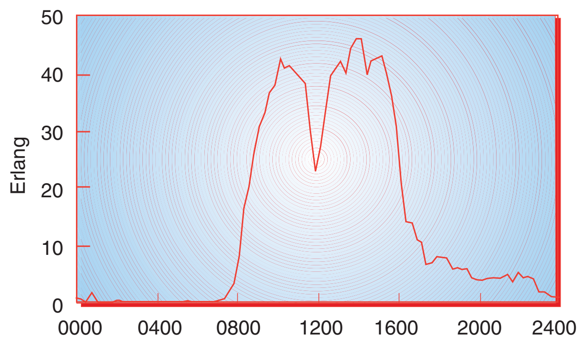


Figure 5 Outgoing traffic (average of weekdays during Period 1), Group 3

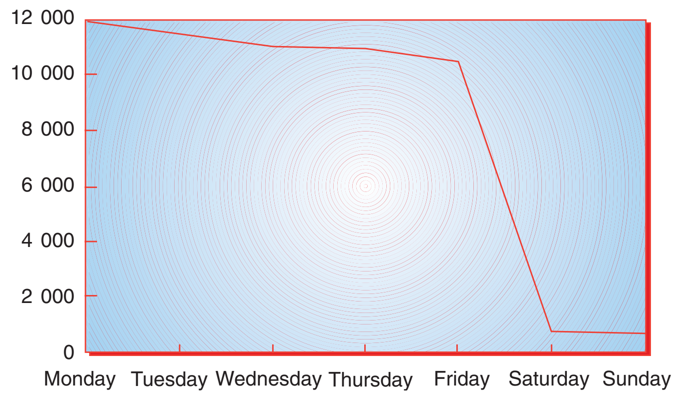


Figure 6 Mean number of outgoing calls per day (Period 1)

Table 2 Number of outgoing calls per weekday divided into categories

	Period 1			Period 2		
Category	**Mean**	**Std.dev.**	**Share**	**Mean**	**Std.dev.**	**Share**
Local	6604.5	375.2	58.9 %	6994.6	304.9	59.4 %
Neighbouring counties	90.8	10.9	0.8 %	96.8	15.4	0.8 %
Domestic long-distance	3316.0	243.4	29.6 %	3515.8	213.9	29.8 %
International	637.7	92.8	5.7 %	587.6	71.7	5.0 %
Mobile telephone	106.4	12.8	0.9 %	103.8	9.6	0.9 %
Paging	38.6	12.0	0.3 %	52.0	17.0	0.4 %
Special services	406.2	42.3	3.6 %	414.7	46.8	3.5 %
Toll-free numbers	14.8	4.0	0.1 %	18.8	6.7	0.2 %
	11215.0	**660.5**	**99.9 %**	**11784.1**	**500.1**	**100.0 %**

Table 3 Outgoing traffic (in Erlang) divided into categories

	Period 1			Period 2		
Category	**Mean**	**Std.dev.**	**Share**	**Mean**	**Std.dev.**	**Share**
Local	13.72	0.68	52.0 %	15.08	0.78	53.2 %
Neighbouring counties	0.26	0.05	1.0 %	0.31	0.08	1.1 %
Domestic long-distance	10.21	0.55	38.7 %	10.80	0.38	38.1 %
International	1.63	0.19	6.6 %	1.54	0.20	5.6 %
Mobile telephone	0.20	0.03	0.7 %	0.18	0.04	0.6 %
Paging	0.010	0.00	0.04 %	0.014	0.00	0.05 %
Special services	0.33	0.06	1.2 %	0.32	0.04	1.1 %
Toll-free numbers	0.05	0.03	0.2 %	0.07	0.03	0.2 %
Total	**26.41**	**1.09**	**100.4 %**	**28.35**	**0.96**	**100.0 %**

- Group 1 (the most expensive calls): International calls
- Group 2 (the expensive calls): Domestic long-distance calls, calls to mobile telephone, and paging
- Group 3 (the inexpensive calls): Local calls and calls to neighbouring counties, special services (some of which are free of charge), and toll-free numbers.

Figures 3, 4 and 5 show the 24-hour traffic profiles of these three groups (registered during Period 1). We see that the 24-hour profiles of Group 2 and Group 3 are quite similar. For Group 1 (international calls) we observe a peak between 1500 and 1600, which we believe is due to calls across the Atlantic.

3.2 Week profile

The statistics are taken during Period 1. The mean number of successful outgoing calls per day is shown in Figure 6. The results show that among the weekdays there are small variations, but the number of calls is highest on Monday and lowest on Friday.

A similar profile is observed in terms of the traffic volume. An interesting observation is that while the mean number of calls on holidays (Saturdays and Sundays) is 6.7 % of the average on a weekday, the corresponding measure for traffic volume is 8.3 %, which means that the mean duration of conversations during weekends is about 25 % longer than that on a weekday. Later (3.5) we shall confirm this when call holding times are discussed.

Table 4 Number of calls and traffic to the other Norwegian universities (S.D. = standard deviation)

To	Period 1						Period 2					
Univ.	Number of calls			Traffic			Number of calls			Traffic		
of	Mean	S.D.	share	Mean	S.D.	share	Mean	S.D.	share	Mean	S.D.	share
Oslo	72.4	12.1	0.65 %	0.193	0.044	0.73 %	95.0	18.5	0.80 %	0.241	0.039	0.84 %
Bergen	43.8	10.5	0.39 %	0.153	0.056	0.58 %	44.2	8.5	0.37 %	0.131	0.030	0.46 %
Tromsø	29.0	5.0	0.26 %	0.092	0.037	0.35 %	29.4	4.6	0.25 %	0.083	0.029	0.29 %
Total	**145.2**		**1.30 %**	**0.438**		**1.66 %**	**168.6**		**1.42 %**	**0.455**		**1.59 %**

Table 5 Number of telefax calls and traffic to the other Norwegian universities
S.D. = standard deviation
share = percentage of fax calls (or traffic by these calls) among the total number of calls (or the total traffic) to the respective universities

To	Period 1						Period 2					
Univ.	Number of calls			Traffic			Number of calls			Traffic		
of	Mean	S.D.	share	Mean	S.D.	share	Mean	S.D.	share	Mean	S.D.	share
Oslo	7.0	3.8	9.7 %	0.009	0.007	4.6 %	6.9	3.1	7.3 %	0.014	0.007	5.8 %
Bergen	3.7	2.3	8.4 %	0.006	0.004	3.9 %	3.5	3.7	7.9 %	0.004	0.004	3.0 %
Tromsø	2.8	1.8	10.0 %	0.004	0.004	5.4 %	4.3	2.5	14.6 %	0.009	0.010	10.8 %
Total	**13.5**		**9.4 %**	**0.019**		**4.6 %**	**14.7**		**8.7 %**	**0.027**		**5.9 %**

3.3 Distribution of service types

The average number of calls and traffic volume (in Erlang) per weekday, registered during two periods and divided into different categories, are shown in Tables 2 and 3.

We observe that the number of calls and the traffic volume have increased 5.1 % and 7.3 % respectively from Period 1 to Period 2. Between these two periods the number of subscribers has only increased by 100 (2 %). The cause of the increase could be that Period 2 is near the end of the semester (and the year), so both NTH and SINTEF use the telephone more than in Period 1.

The distribution between service types does not show any significant changes from Period 1 to Period 2. The exceptions are the use of paging that has increased by more than 30 %, and the number of international calls that has decreased by 6 – 8 %.

The results shows that nearly 60 % of outgoing calls (and 53 % of total outgoing traffic) are local, while around 30 % of outgoing calls (and 40 % of total outgoing traffic) are domestic long-distance calls. The percentage of international calls is about 6 %. The results also show that the duration of long-distance calls is longer than that of local ones (more discussions later in 3.5).

We have also registered the traffic to Oslo and its neighbouring counties. The results show that more that 50 % (the results from the two periods are 51.9 % and 52.2 %) of the domestic long-distance traffic goes to this area, which demonstrates the significance of the connection to Oslo.

There is a fair amount of calls to special services, while there are few calls to toll-free numbers. The reason could be that many of the special services concern business (e.g. asking for telephone number of companies), while the majority of the toll-free calls concern private matters. However, this figure can be changed soon, as companies increasingly establish toll-free telephone services in Norway.

3.4 Traffic to the other Norwegian universities

The mean number of calls and the traffic volume to the other Norwegian universities per weekday are shown in Table 4. Also shown are the shares of the total traffic and the standard deviations.

On average, there are around 150 calls per day to the other Norwegian universities, which accounts for 1.4 % of the total number of outgoing calls and 1.6 % of the total outgoing traffic.

Compared to the registered data from 1989 [4] the traffic to the universities has nearly doubled (from 0.24 Erlang to 0.45 Erlang), although the traffic volume is still low.

3.4.1 Use of telefax

We want to estimate the proportion of telefax calls to the total of outgoing calls. However, the call records do not indicate which numbers are fax machines and which are ordinary telephones. Since we were not allowed to use A-numbers, the only way to register these data was to find all the telefax-numbers at the other Universities (these numbers can be found in their internal telephone directories). Table 5 shows the recorded number of calls and the traffic they generate to fax machines at the other Norwegian universities.

Table 6 Call holding times (in minutes) – Interval 1: Weekdays 0800–1700

	Period 1					Period 2				
Category	**# obs**	**Mean**	**S.D.**	**C.V.**	**Max.**	**# obs**	**Mean**	**S.D.**	**C.V.**	**Max.**
Local	60276	2.92	4.39	1.50	263.4	63355	3.03	4.48	1.48	209.8
Neighb.counties	843	3.93	4.84	1.23	37.9	911	4.40	5.86	1.33	49.3
Long distance	31584	4.15	6.34	1.53	167.9	33660	4.21	6.53	1.55	137.9
Universities	1417	4.28	6.43	1.50	51.3	1660	3.89	5.94	1.53	59.2
Univ. (fax)	118	2.18	2.22	1.02	14.2	140	2.67	3.80	1.44	30.7
International	5880	3.51	5.72	1.63	88.1	5299	3.65	6.10	1.67	71.1
Mobile	996	2.64	3.93	1.49	33.3	958	2.42	4.01	1.66	45.9
Paging	367	0.38	0.21	0.53	2.0	460	0.37	0.22	0.59	2.0
Special serv.	3635	1.10	2.11	1.92	40.8	3636	1.06	1.83	1.73	46.2
Toll-free	136	4.04	4.86	1.20	38.3	161	4.07	4.71	1.16	24.9
Total	**103717**	**3.26**	**5.13**	**1.57**	**263.4**	**108440**	**3.36**	**5.28**	**1.57**	**209.8**

3.5 Distribution of call holding times

The mean call holding time can be calculated by summing up the total traffic volume over the observation period and divide it by the total number of successful calls. However, we can only calculate the mean call holding time for outgoing calls because we cannot measure the number of successful incoming calls. The mean call holding time registered in Period 1 and Period 2 are 3.42 and 3.50 minutes, respectively.

We have also observed the distribution of call holding times. Since call holding times depend on day of the week, time of day, type of call, etc., we have divided the observation periods (Periods 1 and 2) into 3 intervals:

Interval 1:
Weekdays between 0800 and 1700.
In this interval the majority of the conversations are about business matters.

Interval 2:
Weekdays between 1700 and 0800.
In this interval the percentage of private calls is significantly larger, and for most of the call categories the price is lower.

Interval 3:
Weekend.
Many private calls, and the prices are usually the same as in Interval 2.

For each of the intervals we register the mean call holding time, variance, coefficient of variation, and maximum call holding time for every category. The distributions of call holding times for these categories are also shown. The results are presented in sections 3.5.1 – 3.5.3, followed by a comparison and discussion of the results in section 3.5.4.

3.5.1 Interval 1: Weekdays between 0800 and 1700

About 90 % of the calls are made in this interval. The majority of the calls are business conversations. Figure 7 is a histogram showing the distribution of call holding times for all outgoing calls in this interval. Note that in Figure 7 and all the other histograms we have accumulated the longest calls in one single bar to avoid a long tail on the distribution. However, we still use the real distributions in our calculations.

Figure 7 shows a clear concentration of short holding times. The mean holding time (3.26 min.) is almost identical to the registrations made in 1989 [4][1] which show a mean holding time of 3.27 min.

In teletraffic theory a common assumption is that the distribution of call holding times is negative exponential, where the mean value is equal to the standard deviation. Figure 7 shows that the distribution of holding times is more scattered than a negative exponential distribution, since the standard deviation (5.13 min.) is much larger than the mean (3.26 min.). The coefficient of variation, which is the ratio of standard deviation and mean, is 1.57 in this case, while for negative exponential distributions this coefficient should be 1.

Table 6 shows numbers of observations (# obs.), mean values (Mean), standard deviations (S.D.), coefficients of variation (C.V.), and maximum holding times (Max.) for all successful outgoing calls and for each category, from both the observation periods.

The results show that in most of the service categories (except paging and fax) the standard deviations are much larger than the means. The most extreme case appears in the category of special services, where the standard deviation is almost twice as large as the mean.

When comparing the call holding times of those made to the other universities with the average long distance calls, no clear differences were observed.

We observed clear differences in the call holding times between categories, and the mean holding time had a slight increase from Period 1 to Period 2. Table 6 also shows that in most of the categories with "normal" telephone conversations the coefficients of variation (C.D.) are close to 1.5, except for international calls where the C.D. is larger

[1] *There is a little difference in the registration periods: in [4] the registrations were made between 0800 and 1500.*

Table 7 Call holding times (in minutes) – Interval 2: Weekdays 1700–0800
a The charging data files do not show the actual holding times if they are longer than 600 minutes (10 hours). However, TELMAX will generate their own reports if such long conversations occur

	Period 1					Period 2				
Category	**# obs**	**Mean**	**S.D.**	**C.V.**	**Max.**	**# obs**	**Mean**	**S.D.**	**C.V.**	**Max.**
Local	5769	3.71	7.67	2.07	164.7	6591	3.81	8.41	2.21	191.5
Neighb.counties	65	6.33	7.29	1.15	32.1	57	7.29	7.26	1.00	33.8
Long distance	1576	10.05	30.34	3.02	600.0[a]	1498	9.30	11.96	1.29	91.8
Universities	35	7.10	12.15	1.71	53.3	26	3.73	8.26	2.21	40.2
Univ. (fax)	17	1.53	1.22	0.80	5.0	7	2.11	2.18	1.03	6.8
International	497	5.81	9.97	1.72	110.0	577	5.91	10.67	1.81	96.1
Mobile	68	3.16	4.61	1.46	19.0	80	2.62	4.81	1.84	22.7
Paging	19	0.39	0.21	0.54	0.8	60	0.38	0.12	0.32	0.9
Special serv.	427	1.67	3.66	2.19	32.7	511	1.57	3.85	2.45	50.1
Toll-free	12	11.40	13.89	1.22	44.2	27	12.07	12.07	1.00	43.7
Total	**8433**	**4.93**	**15.04**	**3.05**	**600.0**	**9401**	**4.70**	**9.27**	**1.97**	**191.5**

(1.6 – 1.7), and for calls to neighbouring counties where the C.D. is smaller (1.3, with few samples). It is also quite natural that the categories of Fax, Paging, Special services and Toll-free numbers have their own special characteristics.

3.5.2 Interval 2: Weekdays between 1700 and 0800

This interval is outside the normal working hours, and the number of calls is much lower – less than 10 % of that during Interval 1, although the observation time is longer.

Figure 8 shows the distribution of call holding times for all outgoing calls in Interval 2 (Period 1), in the same way as that for Interval 1 (Figure 7).

Like Figure 7, Figure 8 shows a clear concentration of holding times in the lower part of the histogram. Figure 8 shows that the call holding time is even more scatted in Interval 2 than that in Interval 1. The standard deviation (15.04 min.) is more than 3 times as large as the mean (4.93 min.).

The mean holding time is, as expected, clearly longer than that in Interval 1, due to less time pressure and (for most of the categories) lower price.

Table 7 shows numbers of observations, mean values, standard deviations, coefficients of variation, and maximum holding times for all successful outgoing calls and for each category, for both of the observation periods.

Since the number of observations is much lower in this interval than that in Interval 1, the results shown in Table 7 are probably more influenced by statistical fluctuations. For example in Period 1 the single conversation which has the longest holding time (registered with a

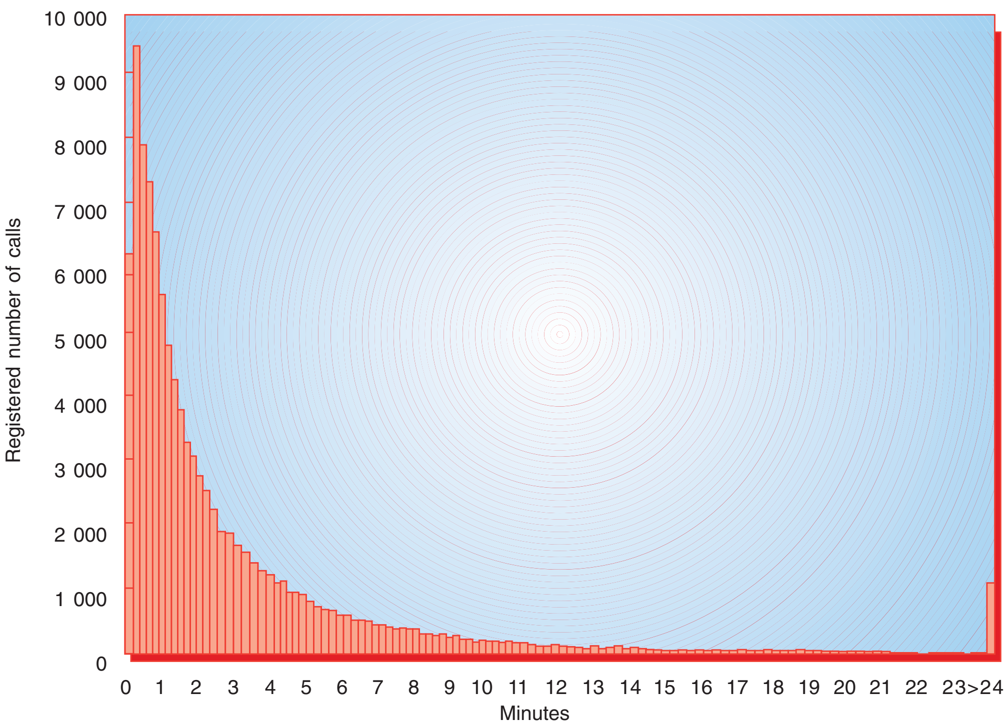


Figure 7 Call holding times, Period 1, Interval 1
Number of observations: 103717 Mean: 3.26 mins
Standard deviation: 5.13 mins Maximum holding time: 263.4 mins

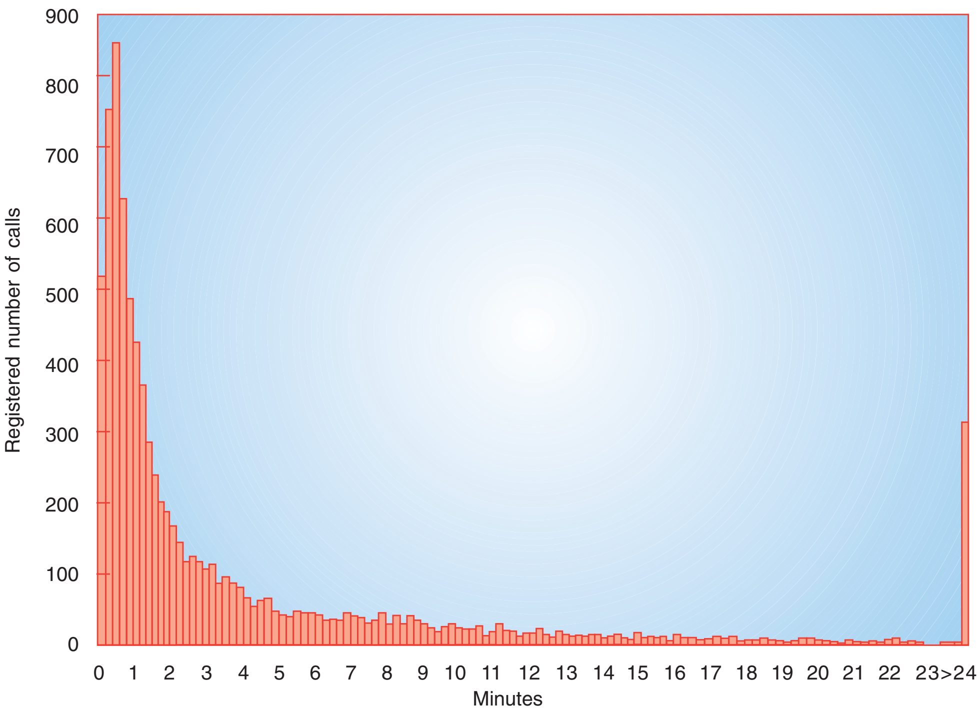


Figure 8 Call holding times, Period 1, Interval 2

Number of observations:	*8433*	*Mean:*	*4.93 mins*
Standard deviation:	*15.04 mins*	*Maximum holding time:*	*600.0 mins*

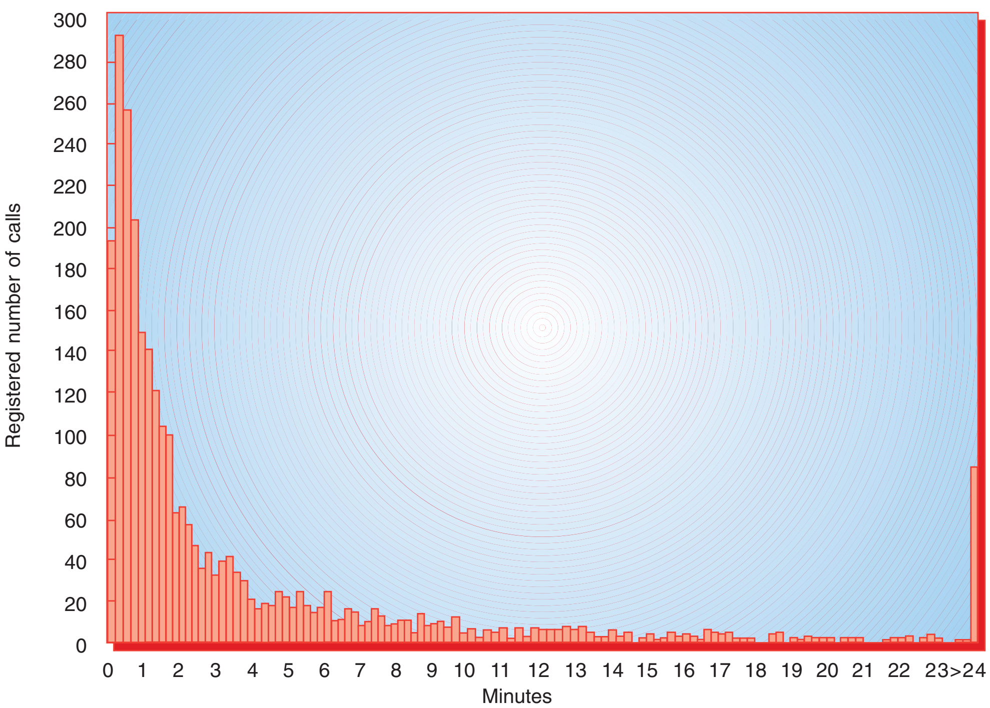


Figure 9 Call holding times, Period 1, Interval 3

Number of observations:	*2799*	*Mean:*	*4.49 mins*
Standard deviation:	*14.11 mins*	*Maximum holding time:*	*518.9 mins*

holding time of 600 min. – see footnote after Table 7) contributes to 13 % increase in the coefficient of variation and 4 % increase in the mean alone.

However, the results still indicate clearly that standard deviation is larger than mean in nearly every category, which confirms that the distributions of holding times do not follow negative exponential distributions.

Table 7 also shows that in contrast to Interval 1, the mean holding time had been reduced by 4.7 % from Period 1 to Period 2 (from 4.93 min. down to 4.70 min.).

When comparing the results from Interval 1 with that from Interval 2, we observed that both means and coefficients of variation are larger (in some cases considerably) outside the normal working hours than that during working hours.

Another observation we made is that among the "normal" domestic calls (i.e. the categories of Local, Neighboring counties, and Long distance calls) the distance plays an important role – the longer the distance is, the longer is mean call holding time. We believe that the reason for this is that users appear to pay more attention to the price when they make more expensive calls, so they choose to make long calls when the price is lower.

3.5.3 Interval 3: Weekends

In Interval 3 both the call intensity and the number of calls are lower than that in Interval 2.

The distribution of call holding times for all outgoing calls (Interval 3, Period 1) is shown in Figure 9.

Figure 9 looks similar to that of Figure 8 (Interval 2). The mean holding time is much longer than that in Interval 1, but a bit shorter than that in Interval 2.

Table 8 shows the number of observations, mean holding times, standard deviations, coefficients of variation, and maximum holding times for all successful outgoing calls and for each category, for both observation periods.

3.5.4 Comparisons and discussions

The registered holding times show that the mean holding time for all calls is considerably longer outside normal working hours. The most significant differences appear on longer distance calls (calls to neighbouring counties, long distance calls

Table 8 *Call holding times (in minutes) – Interval 3: Saturdays and Sundays*

Category	Period 1					Period 2				
	# obs	Mean	S.D.	C.V.	Max.	# obs	Mean	S.D.	C.V.	Max.
Local	1974	3.52	7.01	1.99	114.1	3023	3.64	6.79	1.87	130.9
Neighb.counties	18	4.93	8.13	1.65	27.8	35	5.36	7.28	1.36	36.7
Long distance	424	10.43	31.74	3.02	518.9	619	9.53	13.02	1.37	145.9
Universities	6	1.92	1.84	0.96	5.4	8	3.80	4.39	1.16	13.9
Univ. (fax)	3	2.18	2.75	1.26	5.4	3	4.12	1.21	0.29	5.3
International	139	5.77	9.44	1.64	56.2	135	9.45	19.22	2.03	150.7
Mobile	29	3.45	5.46	1.58	25.4	19	1.56	1.02	0.65	3.4
Paging	17	0.32	0.14	0.44	0.7	54	0.35	0.13	0.37	0.7
Special serv.	196	0.90	1.29	1.43	7.5	208	1.07	1.99	1.86	20.0
Toll-free	2	10.20	4.46	0.44	13.4	1	6.15	-	-	6.15
Total	**2799**	**4.49**	**14.11**	**3.14**	**518.9**	**4094**	**4.56**	**8.86**	**1.94**	**150.7**

– both domestic and international). We believe that the price is the most important factor here, but not the only one.

The results also show large differences between mean holding times for different categories. Paging is the service with the shortest holding time, while special services have the second shortest holding time. Calls to mobile telephones and fax also have relatively short duration. Mean holding time for calls to Toll-free numbers is close to that for long distance domestic calls.

3.6 Other results

3.6.1 Number of call attempts and relation between incoming and outgoing traffic

Number of call attempts of both incoming and outgoing calls were registered by using TELMAX in Period 3 (25.11.91 – 29.11.91) between 0700 and 1900. By comparing the statistics registered by TELMAX with the charging files from the same period, we also found the number of successful outgoing calls[2], and the incoming and outgoing traffic.

Figure 10 shows the average number of call attempts, and Figure 11 shows the incoming and outgoing traffic, both registered per 15 minute intervals, in Period 3 between 0700 and 1900. The total number of call attempts, the percentage of outgoing calls being answered, and the traffic are shown in Table 9.

Figure 10 and Figure 11 show that both in terms of the number of call attempts and the traffic intensity, the 24-hour profiles for incoming and outgoing calls are quite similar.

Table 9 shows that the number of outgoing call attempts is approximately 40 % larger than that of incoming call attempts. Figure 10 shows that this is true for the whole observation period. However, the traffic registered during the same period actually shows that the outgoing traffic is less than the incoming traffic.

There are two possible explanations to this phenomena:

1 outgoing calls have a lower success rate than that for incoming calls

2 mean call holding time for outgoing calls is shorter than that for incoming calls.

Since we neither could register call holding time nor success rate for incoming calls, we cannot give any final conclusion. But we believe that both explanations could be true.

We have reason to believe that the success rate for incoming calls is higher than that for outgoing calls, since the difference between outgoing and incoming call attempts is as big as 40 %, while the reg-

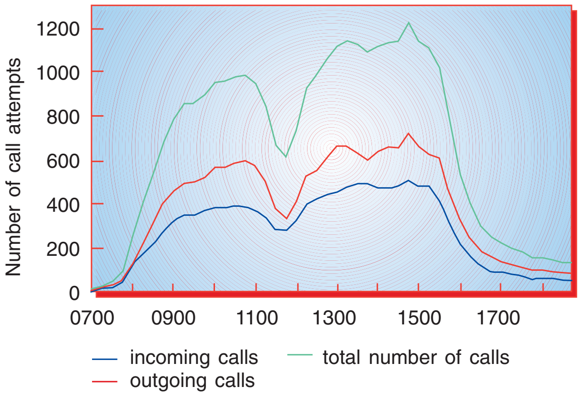


Figure 10 Number of call attempts per 15 minute intervals

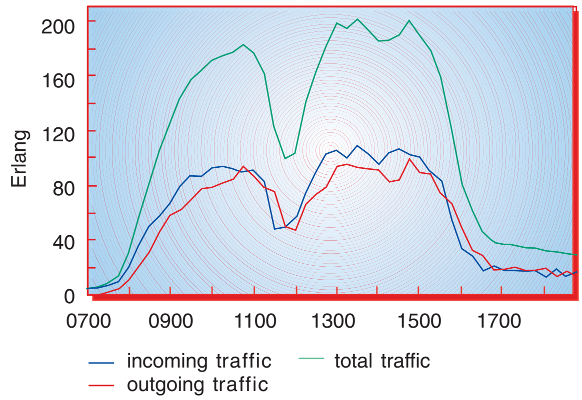


Figure 11 Traffic per 15 minute intervals

[2] *Due to the limitations of the registration tools we were unable to register the number of successful incoming calls.*

Table 9 *Number of call attempts, successful rate for outgoing calls, and traffic*

	Outgoing	Incoming	Total
Number of call attempts (percentage)	93278 (58.4 %)	66431 (41.6 %)	159709 (100 %)
Number of successful calls (success rate)	57438 (61.6 %)	(N/A)	(N/A)
Traffic (percentage)	54.0 (47.6 %)	59.4 (52.4 %)	113.4 (100 %)

istrations from 1989 [4] show that the number of successful outgoing calls was only about 10 % more than that for incoming calls[3]. The reasons for the first explanation are:

- incoming call attempts are not registered until they have reached the main exchange in UNIT/SINTEF, which has virtually no blocking
- the exchange has advanced forwarding functions
- all outgoing call attempts are registered before they reach the public network
- outgoing calls are made to an inhomogeneous group of users, which on average has a higher probability of busy or no answer.

One reason that outgoing calls have shorter holding times is that users tend to make shorter calls when they know that their telephone calls are registered internally. (When the METRO-system was installed, a reduction in mean call holding was observed). Another reason could be that many users outside UNIT/SINTEF do not have forwarding functions, so several short calls (attempts) could be made instead of one long call (which uses forwarding).

[3] *To be correct we should notice that RiT was connected to the exchange in 1989, but not to the present one.*

3.6.2 Use of switchboard operators

The charging files register whether a successful outgoing call is set up manually by switchboard operators. The number of such calls is then compared with the total number of (successful, outgoing) calls during the same periods (Periods 1 and 2). The results show that less than 1 % of these calls have been set up manually. In Period 1, only *836* out of *114,949* were set up manually, while the corresponding numbers in Period 2 are *824* and *121,935.* The result is a bit surprising, since the percentage was considerably higher (2.7 %) in an earlier study made only 2 years before (in 1989 [4]). It indicates that the use of switchboard operators had been very much reduced from 1989 to 1991.

For incoming calls the registration was based on the number of call attempts. The registration took place in Period 4 (Jan. 13 – 14 1992). A total of 31,693 incoming call attempts were registered, and among them 3,323 (i.e. 10.5 %) were set up by switchboard operators. In other words, 89.5 % of call attempts were made directly without use of switchboard operators. The observations made in 1989 [4] show that 82.6 % of incoming call attempts used direct dialling. Thus, even the incoming traffic load on operators has been reduced a lot.

When comparing our results with the results from 1989 [4] we should notice that the Regional Hospital of Trondheim (RiT) was connected to the exchange in 1989, but not since January 1990. It is likely that a larger portion of the calls made to RiT use switchboard operators than that to UNIT and SINTEF. Apart from that there is reason to believe that better information and a more convenient system from users' point of view contribute to the reduced need for manual assistance.

3.6.3 Traffic concentration

Traffic concentration is a measure of traffic variations over time.

Before we start discussing the traffic concentration, we first take a look at the *Busy hour*.

The busy hour is defined as the 4 consecutive 15 minute periods during a day with the highest traffic.

We have calculated busy hour (for outgoing traffic) for each weekday in Periods 1 and 2. The results are shown in Table 10. In the table we also show the *Time consistent busy hour,* which is defined in a similar way as the busy hour, but the traffic in each period is the *average* over all days of the observation period.

Table 10 shows that the busy hour without exception appears after lunch. The time consistent busy hour for Periods 1 and 2 coincide, and the busy hour for each day starts between 1300 and 1400. The traffic intensity during the busy hour follows the same pattern as for the average traffic intensity during a day: Most traffic in the beginning of a week, and least traffic on Friday. There is also a clear increase of traffic during busy hour from Period 1 to Period 2.

Based on the available data the busy hour for the total two-way traffic in Period 3 (Nov. 25 – 29 1991 between 0700 and 1900) have also been calculated. The results are shown in Table 11. It shows that the busy hour for total traffic occurs at approximately the same intervals, i.e. starting at between 1300 and 1400.

Now we can give the definition of the *Traffic concentration,* which is the relation between the mean traffic in the busy hour and the mean traffic in the 24 hour

Table 10 *The busy hour for outgoing traffic*

	Period 1 (Oct 7 – 20, 1991)		Period 2 (Nov 25 – Dec 8, 1991)	
Day of week	Traffic	Time interval	Traffic	Time interval
Monday	90.433	1400 – 1500	95.227	1430 – 1530
Tuesday	84.694	1415 – 1515	96.139	1345 – 1445
Wednesday	85.547	1345 – 1445	93.646	1300 – 1400
Thursday	84.687	1315 – 1415	92.512	1315 – 1415
Friday	83.586	1315 – 1415	91.356	1430 – 1530
Time consistent busy hour	83.715	1315 – 1415	92.205	1315 – 1415

period the busy hour belongs to, averaged over all weekdays in the observation period.

Since we did not have registered data for the total traffic over 24-hour periods, we again have to make estimates based on the outgoing traffic. Fortunately, Table 10, Table 11 and Figure 11 indicate that the estimates are representative for the total traffic.

The traffic concentration in Periods 1 and 2 are 3.24 and 3.31 respectively, which means that on weekdays the carried traffic during the busy hour is more than three times as high as the mean.

4 Conclusion

This study has been especially focused on analysis of the external outgoing traffic. The total traffic and incoming call attempts have also been studied. The analysis cover call attempts, traffic, and holding times. Variations over time, statistical distributions and distribution of traffic over categories have also been studied. An important key to this study is the analysis based on the numbers called.

A more detailed study on the use of additional special services was planned, but it could not be performed because the MD110 lacks the necessary functions. In this field the old 10R exchange was better equipped with statistics counters. Another study we wanted to perform was the analysis of internal traffic. However, we were unable to collect statistics in order to establish a traffic matrix between the LIMs[4]. We therefore decided not to do any analysis of internal traffic in this study.

This study has been quite extensive. The results could be useful for optimal planning and management. By observation and analysis of traffic, a yearly saving of *120,000* kroners has already been achieved without any reduction of service. A further follow-up of observation results has the potential to achieve both cost saving and improvement of service qualities. We hope that the analysis done in this study can give some contributions to this.

The present study is done within the particular environment of a university/research institute. It may not in all aspects be representative for any large user environments. Still, we believe that it is possible to extract knowledge and insight of value also for business communication in general.

References

1 Myskja, A. A recording and processing system for accounting and traffic analysis on a large PABX. *IEEE transactions on communication technology,* COM-19(5), 1971.

2 Myskja. A. Early experience with an SPC PABX with new facilities. I: *International conference on private electronic switching systems,* April 1978.

3 Jensen, E B, Myskja, A. Metaconta 10R : the telephone exchange at the University of Trondheim (in Norwegian). *Telektronikk,* 74(4), 1978.

4 Heegaard, P E. *A traffic study of the telephone exchange at the University of Trondheim* (in Norwegian). Trondheim, ELAB, 1990. (Report STF44 F90006.)

5 Iversen, V B. *Data and teletraffic theory, vol. 1* (in Danish). Den private Ingeniørfond, 1985.

6 Feng, B, Myskja, A, Heegaard, P E. *Analysis of external traffic at UNIT/SINTEF's telephone exchange, MD110* (in Norwegian). Trondheim, SINTEF DELAB, 1992. (Report STF40 A92047.)

Table 11 The busy hour for total traffic, Period 3

Day of week	Traffic	Time interval
Monday 25.11	203.84	1300 – 1400
Tuesday 26.11	200.78	1300 – 1400
Wednesday 27.11	189.50	1400 – 1500
Thursday 28.11	194.94	1300 – 1400
Friday 29.11	198.58	1300 – 1400
Time consistent busy hour	197.07	1300 – 1400

[4] *It is possible to measure traffic to or from each of the 36 LIMs, but they do not give any information about traffic between any pairs of LIMs.*

LAN Interconnection Traffic Measurements

BY SIGMUND GAAREN

1 Introduction

Interconnection of geographically separated LANs (Local Area Networks) is expected to be one of the most important applications for existing and future public data networks. This article presents the results from LAN interconnections traffic measurements at a selection of 12 different LANs; nine customers' sites and three Telenor sites.

Information about the data traffic flowing in and out of LANs is needed for optimizing the WAN (Wide Area Network) capacity usage, and, hence, reduce customers' LAN-LAN transmission costs. The choice of appropriate services and capacities for LAN interconnections should be aided by *measurements* in order to tell the customer exactly which product that best suits each specific LAN communications demand.

Previous LAN traffic analyses have concentrated on the dynamics of internal data flow on universities' and research laboratories' local networks. The measurements in this article were conducted from January to May 1994, and will provide information about the data traffic for the heaviest users of LAN interconnection services in Norway.

The main objectives for this investigation of the data traffic characteristics on LAN interconnections were:

- Identification and description of LAN *categories* and their demands for WAN transmission capabilities
- Recognition of the LAN interconnection *protocols,* on OSI layers 3 to 7, in use; and their per centage occurrences and frame sizes
- Evaluation of the LAN interconnection *utilization* and traffic patterns on different time scales.

List of abbreviations and acronyms

ATM	Asynchronous Transfer Mode
BPS	Bits Per Second
CAD	Computer Aided Design
CMIP	Communications Management Information Protocol
CoV	Coefficient of Variation
FDDI	Fiber Distributed Data Interface
FM	Foundation Manager
FTP	File Transfer Protocol
IAT	InterArrival Time
ICMP	Internet Control Message Protocol
IGRP	Interior Gateway Routing Protocol
IP	Internet Protocol
IPX	Internetwork Packet eXchange
ISO	International Standards Organization
LAN	Local Area Network
LL	Leased Lines
LLC	Logical Link Control
MAC	Medium Access Control
ms	milliseconds
NCP	Netware Core Protocol
NetBEUI	NetBIOS Extended User Interface
NetBIOS	Network Basic Input Output System
NFS	Network File System
OS	Operating System
OSI	Open Systems Interconnect
OSPF	Open Shortest Path First
PC	Personal Computer
PPP	Point-to-Point Protocol
RIP	Routing Information Protocol
SDLC	Synchronous Data Link Control
SMTP	Simple Mail Transfer Protocol
SNA	System Network Architecture
SNMP	Simple Network Management Protocol
SPX	Sequenced Packet eXchange
TCP	Transmission Control Protocol
TMN	Telecommunications Management Network
TR	Token Ring
UDP	User Datagram Protocol
WAN	Wide Area Network
XNS	Xerox Network Systems
XWin	XWindows

2 Measurement technique and set-up

This section will provide a brief introduction to the measurement technicalities and set-up, and the limitations of the measurement procedure. For more detailed information about the measurement methodology, please consult [2].

2.1 Definitions and requirements

The LAN as a whole is defined as the source and sink for the traffic for the traffic coming in to and going out of the LAN. In order to understand the LANs as source and sink for data traffic, information about applications, usage and location of shared facilities (e.g. servers) were obtained by interviews with LAN managers.

The following metrics were measured:

- bits per second
- frames per second
- frame length in bytes
- frame interarrival time (IAT) in milliseconds
- per centage external traffic relative to total internal LAN traffic
- per centage occurrence of 24 LAN-protocols on OSI layer 3, 4 and 7
- number of erroneous frames.

The frame interarrival time is defined as the time gap between arrivals of two adjacent frames. All the metrics are further explained in [2]. Figure 4.1 in Section 4.1 contains all detected protocols.

A data traffic file was generated per day and per direction. WAN-links are offered with bi-directional data transfer capabilities; therefore, one file per traffic direction, LAN incoming and LAN outgoing traffic, was required. Also, protocol summary and frame size distribution files were generated once per day.

The protocol analyser should not corrupt the transmitted data, generate any traffic, or lose any frames.

2.2 Tool and observation point

Since the LAN interconnection traffic data were captured at many different types of networks for shorter periods of time, an IP software monitor, e.g. *nnstat, etherfind* or *tcpdump,* was considered unsuitable for this purpose.

Due to the limitations in OSI layer 4 and 7 decoding, interarrival time detection, and separation of incoming and outgoing traffic in available WAN analysers, a LAN protocol analyser was employed. The chosen protocol analyser is a Foundation Manager from ProTools, which is an OS/2 portable computer, fitted with a specially designed traffic measurement software, and Ethernet and Token Ring interfaces. Figure 2.1 illustrates the observation point from which the data traffic was captured at customers' sites.

Since the external traffic is bound to go through the router, the LAN interconnection traffic was extracted by filtering out all other MAC frames except the frames to and from the router. All frames with the router's MAC address as source is defined as *incoming* traffic, depicted as 'in' in Figure 2.1. Consequently, all frames where the router is destination are identified as *outgoing* traffic.

These definitions assume that the internal traffic on the LAN does not go via the router; which is usually true, unless the router also acts as an internal server or several LAN segments are connected directly to the router. In the latter case, filter functions in the protocol analyser can be utilized to extract the WAN traffic, see [2].

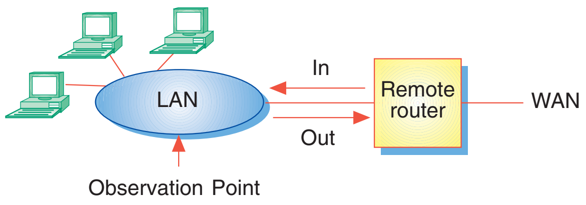


Figure 2.1 Observation point for LAN interconnection measurements

2.3 Technical constraints

The fact that the observation point for these measurements is on the LAN side of the router, as illustrated in Figure 2.1, yields that the measurement data obtained cannot represent the WAN link traffic exactly. Also, the *actual* WAN link traffic will be less than the *theoretical* WAN link capacity. These two issues are due to the parameters listed below:

- The LAN protocol overhead. Ethernet has 18 bytes overhead, plus a padding to ensure a minimum frame size of 64 bytes; the maximum data field (payload) size being 1500 bytes. Token Ring has 20 bytes overhead, but no minimum frame size requirement; the token hold time limits the maximum frame size.
- The WAN protocols overhead. The most common protocols on OSI layer 2 are the SDLC family, with 6 bytes overhead and unlimited data field; and the Point-to-Point Protocol (PPP), which has 8 bytes overhead and 1500 bytes as maximum data field. X.25, a layer 3 protocol, adds another 3 bytes to the layer 2 protocol it is superimposed over; and the data field being maximum 128 or 1024 bytes. LAN data fields that exceed the WAN data field size will be split into several WAN frames. WAN protocols can also be router specific.
- The router configuration, where the buffer sizes, the encapsulation of LAN data fields into WAN frames, and the encapsulation of WAN data fields into LAN frames are of particular interest.
- The WAN window size, which restricts the number of unacknowledged frames or packets that the receiving end can buffer up.
- The control frames in a WAN protocol, which will steal capacity from the payload.
- Routing information interchanged among routers on the WAN link, using protocols such as RIP, IGRP and OSPF, is not visible from the observation point.
- LAN broadcast containing the router address will be measured, but will not be transmitted on the WAN link. This has proven to be of little significance in the measurements conducted here.

Since all these parameters will vary from case to case, it is *not* possible to calculate the exact WAN link utilization for each LAN interconnection based on the obtained measurement data; and it is *not* the objective of this study. The objective is to identify the LAN sites as traffic sources to the WAN links, and therefore, we do not wish to look into the different mechanisms of WAN protocols, which can be proprietary router specific protocols.

Figure 2.1 illustrates the simplified topological situation, where the external LAN traffic must be transmitted over the single WAN-link. More complex LAN and WAN topologies may complicate the interpretation of the measurement data obtained. [2] provides examples of such complex topologies and how to deal with them.

Since the filtering of the traffic to and from the router is done by use of the MAC-address, these measurements are restricted to routed networks. Bridged networks use end-to-end MAC-addressing, and these measurements are therefore unsuited for such networks.

2.4 Some considerations about the integration time

The *integration time* or *integration period* is defined as the time period over which measurement data are collected. When the integration time is completed, the data obtained in that time period are stored, data registers are reset, and a new integration time begins. Introducing an integration time yields statistical limitations as all the figures will be accumulated over that time period. Taking the average over the integration time will hide extreme values of the metrics given in Section 2.1.

Hence, the ability to reproduce bursty traffic behaviour is dependent on the integration time: the shorter the integration time is, the better the traffic description becomes. Examples in [2] show that the highest traffic load peaks are reduced to about half the size when the integration time increases from 1 second to 10 seconds. Increasing the integration time further to 1 minute yields yet another halving of the traffic peaks. However, the total number of bytes and average traffic load per hour or day are independent of the integration time.

Clearly, an integration time in the order of microseconds or milliseconds is required to obtain all details about the traffic dynamics. In [6], the interactive 'think and turn-around' time is found to be between 150 and 200 milliseconds; and the time length of a burst is in the order of 100 milliseconds. The maximum end-to-end delay the user can accept is related to the subjective perception of the quality of service.

WAN link measurement tools are capable of managing integration times down to 1 minute, which is insufficient for reproducing the dynamic pattern of the LAN traffic described in the paragraph above.

Choosing the appropriate integration time is a trade off between the reproduction of the exact traffic pattern and the data storage available. The nature of the measurement task, where the protocol analyser is left out at the customers' site, did not give the opportunity to first find the busiest periods and then just store the traffic during these hours, which would reduce the demand for storage capacity. Also, diurnal traffic graphs are of interest, so that the measurements had to take place 24 hours per day. Otherwise, a triggering table could be employed in order to measure traffic in certain periods, e.g. the working hours.

Taking all this into account, an integration time of 10 seconds was chosen. As indicated, a measurement with this integration time will not provide exact indications of the user's instantaneous peak demand for LAN interconnection capacity. This fact should be kept in mind when evaluating the maximum traffic loads in this article. On the other hand, the data measurement files become tractable for the storage media and the SAS graphical and statistical software package.

2.5 Data storage

The data traffic analysis was carried out on a UNIX workstation, and the only possible data transport medium from the OS/2 portable computer was via 1.44 Mbyte floppy discs. This fact limits the maximum file size to the capacity of a floppy disc.

Data capture at customers' sites requires that security precautions are taken against insight to data information, passwords and similar. Raw frame headers, which demand post-processing of measurement data, would require too large storage capacity, therefore, real-time processing of traffic data was chosen.

The processed traffic data was organised in a table; where new data was added every 10th second throughout the measurement duration. An integration time of 10 seconds yields 8640 lines in the table per day per traffic direction.

3 Five LAN categories

Measurements on LAN interconnections were carried out on a selection of 12 LAN sites in 8 different companies. The selection of LAN sites includes business types like public administration, oil exploration, education, research, engineering, construction, and Telenor internal.

The chosen LAN interconnections are *not* an unbiased selection of all LAN interconnections in Norway. This is due to the fact that conduction of these measurements is rather time consuming. In order to get a representative picture of the traffic at each LAN site, one week is allowed per site. When considering the level of significance for the results published in this report, one should keep in mind that this is a *limited* selection of all LAN interconnections in Norway, and businesses like bank, insurance, media, and trade are not included.

However, the narrow group of large companies in the selection represents major traffic generators from various types of LAN interconnection topologies, applications and usage. Hence, these measurements will provide a first and important experience with data traffic over LAN interconnections for different business types.

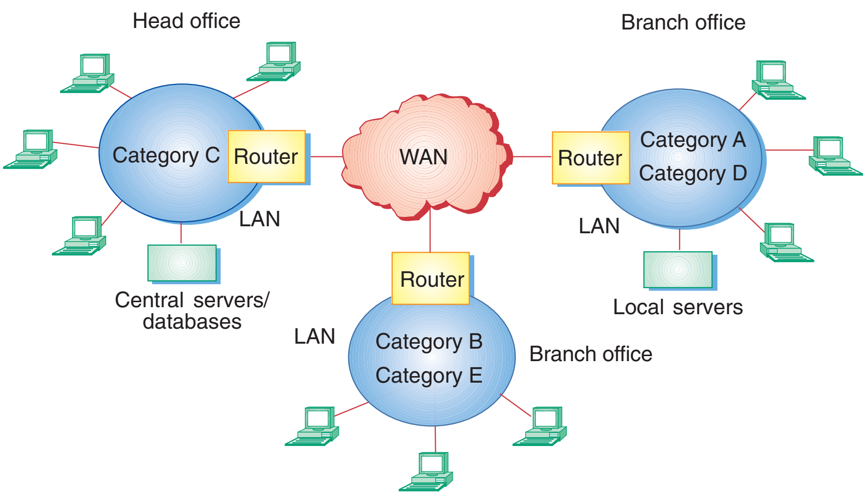


Figure 3.1 Topology overview for the five LAN categories

3.1 Organizing the selection into categories

For a structured approach to the different LAN interconnections under study, their features and characteristics, an organization into *categories* of the measured LAN sites is appropriate. For this categorisation, type of office, location of servers and external applications in use, were important.

The categories are:

- Category A: Branch offices with local servers, external support for database enquiries and central application enquiries.
- Category B: Branch offices without local servers, external support for all applications.
- Category C: Head offices, database responses and central application responses to external branch offices.
- Category D: Internet environment, open and distributed application support worldwide.
- Category E: Computer Aided Design (CAD) networks, external support for graphical client-server applications.

An overview of the five categories is given in Figure 3.1. Categories A and B are branch offices: A with basic office applications, like Word, Excel and similar, served locally; and B with no servers located on the LAN site. These categories can be looked upon as information obtainers since their users access central databases, located in head offices. Category B will have all applications served from a head office.

Category C consists of head offices, serving central database and applications for several branch offices, and such offices may be described as information providers. These first three categories are closed company internal networks. Category D represents an open Internet environment where users can access servers worldwide. This category can be compared to A as regards server locations, but category D LAN sites are not connected to any particular head office as they are part of an open and distributed server environment.

The last category, E, is similar to B in configuration, but is specific to graphical (or CAD) applications, assuming that this usage generates more data traffic than common office packages and database requests and responses.

3.2 LAN interconnection traffic load per category

The LAN interconnection traffic load per LAN category is given in Table 3.1. The figures in the table are given as intervals in order to illustrate the variation between the sites in each category. The features of interest per category are busy hour mean, maximum load, percentage external traffic, traffic direction and ratio.

The *busy hour mean* are given per 100 users, to allow for comparison between LAN sites with different numbers of users. The busy hour time window is investigated further in Section 5.3. The term *users* means number of terminals connected to the LAN site. The measurement tool was *not* specified for recognizing the number of *active* users. Therefore, all terminals that have access to the LAN are counted as users. For networks with a large share of passive users, the weighting by all users will represent an underestimate of the traffic load per user.

The *maximum* load has not been weighted by number of users because traffic peaks usually are caused by a few machines loading the LAN interconnection heavily. For maximum peaks, the considerations about the integration time in Section 2.4 and [2] should be taken into account. *Percentage external traffic* indicates the relative amount of data traffic on the LAN that is generated from or destined for external resources.

As today's WAN services' capacities are symmetric, the *traffic direction* with the heaviest load has to be considered for the dimensioning of the WAN links. In the table, the topical traffic direction, incoming or outgoing, and the in/out or out/in *ratio* of number of bytes carried are indicated.

According to Table 3.1, category A LAN sites have larger incoming than outgoing traffic due to short requests and longer responses from central databases and applications. The category B branch offices, without any servers, have the same traffic pattern as category A, but in larger quantities. The incoming frame lengths are up to 1100 bytes and the outgoing frame lengths are less than 150 bytes for these two categories. Frame interarrival times in A are less than 500 ms, and less than 250 ms for B.

Head offices, in category C, have 2-7 times larger outgoing than incoming traffic, due to the central database and application traffic to and from the branch offices. Since the users in the head office can access the servers locally, the number of users in the branch offices is of more significance for the LAN interconnection traffic than the number of users in the head office. The busy hour mean outgoing traffic for category C is between 2 and 9 kbit/s per 100 users in the branch offices, which is comparable to category A. Incoming frames to the head offices in this selection are less than 200 bytes, and outgoing frames are up to 700 bytes; frame interarrival times are up to 150 ms.

In these measurements, category D's busy hour mean capacity demand per 100 users lies between that of categories A and C (head office users). This is quite reasonable, as category D users are regarded as more advanced than category A users, but do not concentrate as much traffic per 100 users as category C LAN sites do. Category D has less external traffic than category B, since the latter has no local servers. Frame sizes in Internet LAN sites are up to 550 bytes for external traffic; frames are separated in time by up to 300 ms.

The topology and the traffic pattern of the CAD networks in category E are similar to the branch offices in category B, but the CAD applications generate approximately 15-30 times more traffic per 100 users compared to standard office packages in use in category B. For category E, busy hour mean traffic up to 7.2 Mbit/s per 100 users has been detected. (Note that the category E LAN sites contained 15 to 18 users, which makes the busy hour mean weighted per 100 users larger than the unweighted maximum load in Table 3.1.) In the server to client direction, frames are up to 1400 bytes; and in the opposite traffic direction the frames are from 150 to 250 bytes. Frame interarrival times are less than 300 ms.

The categories identify different needs for LAN interconnection capacity. Categories A, C and D generate relatively modest traffic volumes, while categories B and E are more demanding with respect to WAN link transmission capabilities. This is due to the location of servers in the remote head offices for categories B and E. The LANs in these two categories transmit and receive more than 70 % of the data on their LAN sites to external servers, which is illustrated in the 'percentage external traffic per category' column in Table 3.1.

Table 3.1 LAN interconnection traffic loads per category

Category	Number of users	Busy hour mean per 100 users 1)	Maximum load in 10 seconds 2)	Percentage external traffic 3)	Traffic direction, ratio 4)
A	50 – 300	5.3 k – 21 k	60 k – 150 k	< 10 %	In, < 5
B	50 – 150	100 k – 343 k	1.1 M – 1.6 M	> 70 %	In, 1.5 – 4
C *)	280 – 3500 4000 – 6000	22 k – 74 k 2.2 k – 9.4 k	187 k – 2.1 M	< 25 %	Out, 2 – 7
D	250 – 270	21 k – 24 k	387 k – 437 k	15 – 20 %	-, 1.4 – 5
E	15 – 18	4.4 M – 7.2 M	2.7 M – 3.9 M	> 80 %	S-to-C, 2 – 4

Note: k – kbit/s; M – Mbit/s.

**) The first row indicates number of users in the head office, and traffic per 100 head office users. The second row represents number of users in the branch offices, and traffic per 100 branch office users.*

1) Busy hour mean – average traffic over the busiest hour in the measurement period.

2) Maximum – the highest average traffic load for the duration of one integration time period of 10 seconds.

3) Percentage external traffic – ratio of external traffic, both incoming and outgoing, to total traffic on the LAN.

4) Traffic direction, ratio – if outgoing traffic is dominating, the ratio of bytes is outgoing divided by incoming; or if the incoming traffic is dominating, the ratio of bytes is incoming divided by outgoing. S-to-C = Server to Client.

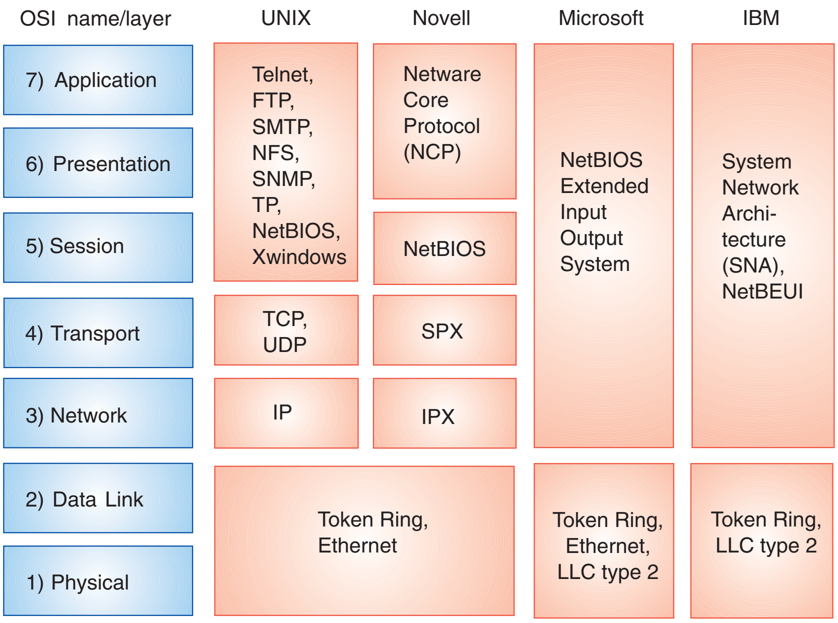


Figure 4.1 Detected protocol stacks [7] [8]

4 Protocols on LAN interconnections

The protocols identified on the LAN interconnections for this selection of LAN sites are presented in this section. Also, the frame sizes per layer 7 protocol are looked into.

4.1 Protocol occurrence

The detected protocol stacks are given in Figure 4.1, in order to visualize the protocols' relations.

The most common protocol on layer 3 is IP, detected at 10 out of 12 LAN sites and carrying over 60 % of the traffic measured totally. Novells IPX were present at one third of the customers' sites, representing approximately 20 % of the total traffic. LLC, indicating IBM SNA applications, counted for a 17 % traffic share and was represented at 5 LAN sites.

Xerox' XNS were in use in CAD environments, but carried only a small fraction of the total traffic quanta. Protocols stemming from Apple, Vines, OSI IP and DECnet are not registered in the external traffic to and from the 12 LAN sites at all.

On layer 4, TCP and UDP are dominating, and only small portions of SPX and OSI TP traffic were detected. The most used protocols on the application layer are Novell's NCP, Telnet over TCP/IP and IBM's NetBEUI. Other IP-related applications as Xwindows, NFS and FTP are also widely in use. SMTP over TCP/IP, NetBIOS over TCP/IP and NetBIOS over IPX only carry a small amount of the total external traffic at the LAN sites.

4.2 Frame sizes per application

The application layer protocols can be separated into interactive and bulk transfer protocols. Telnet and Xwindow are regarded as interactive protocols, which are recognized by short and few frames per transmission. FTP, Xmail and NFS belong to the group denoted as bulk transfer protocols, where long frames are sent back to back until the bulk message has been transmitted. [5] [9]

NetBios and NCP are application packages and may contain several applications, both interactive and bulk transfer.

Telnet generally produces short frames: the average frame size is around 100 bytes, which is as expected for an interactive application, but also longer Telnet frames, around 150 bytes on average, are detected in database responses. Xwindows generally carries more information than Telnet, and its average frame sizes per category varies from 150 to 210 bytes.

NetBios superimposed over IPX and NetBeui generates short frames, typically 100 bytes on average, but NetBios over TCP contains 2.5 times longer frames on average than with the two previous protocols. This may be due to the different configuration of NetBios over the various transport protocols. NCP appears in PC networks with average frame lengths around 220–230 bytes.

In IP-environments, FTP has an average frame size of 400 bytes, which is characteristic for a bulk transfer protocol. However, for PC networks, FTP frames are 160 bytes on average, owing to transmission of shorter files. The other main bulk transfer protocol is NFS, which has an average frame size of 200 bytes for plain office applications. The NFS frames are longer, 430 bytes on average, when CAD files are transferred. For closed company networks, the frame sizes in Xmail (SMTP) are modest: only 90 bytes on average. In the Internet environment, where Xmail is frequently in use, the average frame size is 250 bytes.

Generally, the frames in the bulk transfer protocols FTP and NFS become longer as the traffic volumes transmitted become larger. The frame sizes in the interactive applications Telnet and Xwindows remains relatively short, regardless of external LAN traffic load.

5 The LAN interconnection utilization

This section identifies the LAN interconnections and the traffic load applied to them. An example of traffic loads over various time windows and the LAN traffic's demand to WAN services are given.

5.1 The LAN interconnections

The most popular WAN service for interconnecting this selection of LAN sites is the leased line service. Nine out of 12 sites were interconnected by use of this service, where four of the leased lines were 2 Mbit/s, one was 1024 kbit/s, one was 256 kbit/s, and three were 64 kbit/s.

Two out of 12 LAN sites were connected to an X.25 service, with bit rates of 9.6 and 19.2 kbit/s. A shared medium 100 Mbit/s FDDI was also used as a LAN interconnection, but this type of service can not be compared to the dedicated point-to-point WAN services, such as leased lines and X.25.

5.2 Traffic load applied to the LAN interconnections

This section presents the traffic loads applied to the LAN interconnections, or WAN links, from the selection of LAN sites. Table 5.1 is organised by WAN link type with the busiest hour mean and maximum load expressed as per centage figures of the theoretical WAN link capacity. For the maximum load, the constraints introduced by the 10 second integration time are presented in Section 2.4. For 64 kbit/s and 2 Mbit/s leased lines, there are 3 and 4 LAN interconnections involved, respectively. Therefore, for these two types of WAN links, the figures in the tables are expressed as intervals covering the variation in capacity requirements per LAN interconnection. As WAN links are bi-directional, the traffic direction, incoming or outgoing, with the heaviest load is considered here.

The average LAN interconnection capacity requirement during the busiest hour for 2 Mbit/s represents up to 40 % of the *theoretical* WAN capacity, not taking into account the LAN/WAN transmission issues explained in Section 2.3.

For the LAN site connected to the 1024 kbit/s leased line, the busiest hour generates an outgoing traffic equivalent to 105 % of the theoretical WAN link capacity. This overload traffic situation is illustrated in Figure 5.4 and discussed in Section 5.3. The 10 seconds wide maximum peak is 3.8 times the theoretical WAN link capacity.

Of the WAN links with less capacity than 1 Mbit/s, 64 kbit/s leased lines and 19.2 kbit/s X.25 are heaviest loaded, with up to 89 % of the theoretical WAN link capacity during busy hour. The 10 seconds wide incoming maximum peaks for the X.25 service is over 7 times the theoretical WAN link capacity. This may be due to some buffer mechanisms in the gateways.

Table 5.1 Traffic load applied to the LAN interconnections, expressed as percentage of the WAN capacity

LAN Interconnection	Busiest hour mean	Maximum load in 10 s	Direction
2 M LL	2.5 – 40 %	55 – 123 %	3 In + Out
1024 k LL	105 %	381 %	Out
256 k LL	34.8 %	73.1 %	Out
64 k LL	64.1 – 89.1 %	93.8 – 683 %	2 In + Out
19.2 k X.25	83.3 %	781 %	In
9.6 k X.25	29.2 %	740 %	In
100 M FDDI [1)]	2.6 %	10.5 %	Out

Note: k – kbit/s; M – Mbit/s; LL – Leased Lines.

[1)] The 100 Mbit/s FDDI ring capacity is shared between 5 routers. 20 Mbit/s is estimated as WAN capacity per router, and is used as basis for percentage capacity calculation.

The bandwidth not utilized for LAN-LAN communication is used for telephony.

The traffic load graphs for three days, given in Figures 5.1 and 5.2, represent the external traffic load measured in both directions at the LAN site, and applied to the router for communication over the 1024 kbit/s.

In general, LAN-LAN traffic is extremely bursty, and it is almost impossible to find any repeating pattern at any time scale, as discussed in [3]. However, the three day traffic load periods display a daily repeating pattern. This is due to the different activity level on the LAN-LAN network from day to night. When considering the one day outgoing graph in Figure 5.3, one can see that the outgoing traffic is most intense between 0800 and 1100 hours, between 1230 and 1600 hours, and between 1700 and 1900 hours; but there is *no* particular busy hour. The traffic peaks appear almost randomly from early morning to late evening. During the night, there is an almost constant level of incoming back-up traffic, and hardly any outgoing traffic.

As explained in Section 3.2, the server-to-remote clients traffic is larger than the clients-to-remote server traffic. Since the observation point is in the head office, where the CAD server is situated, and the clients are situated on the remote site, the

5.3 LAN interconnection traffic example over various time windows

This section presents examples of the LAN interconnection traffic load patterns for 3 days, 1 day and 1 hour, respectively, for a remote client to server CAD environment (category E). The traffic loads over the different time intervals are shown in Figure 5.1 through 5.4. This particular LAN site is interconnected with a 1024 kbit/s data channel, multiplexed on a standard 2 Mbit/s leased line.

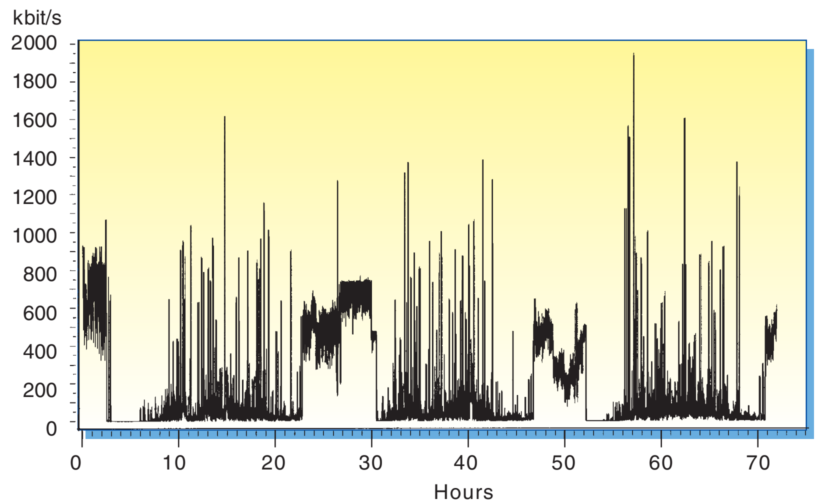


Figure 5.1 Incoming traffic load during three days, category E

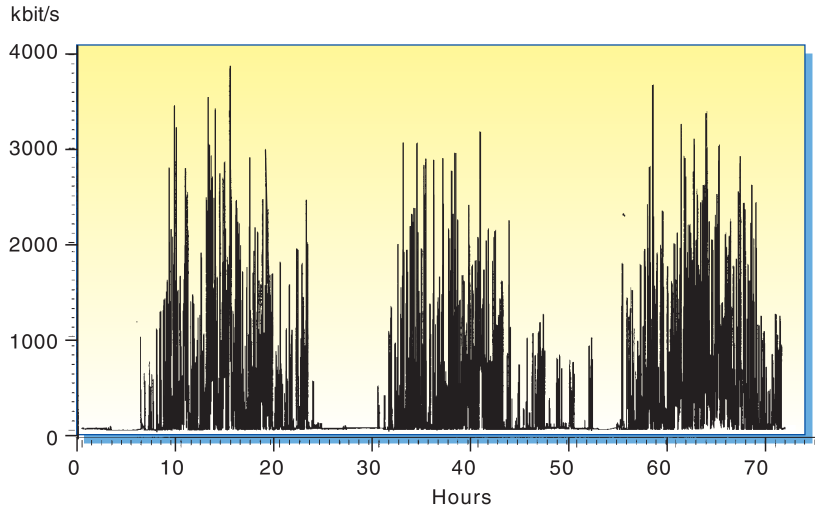


Figure 5.2 Outgoing traffic load during three days, category E

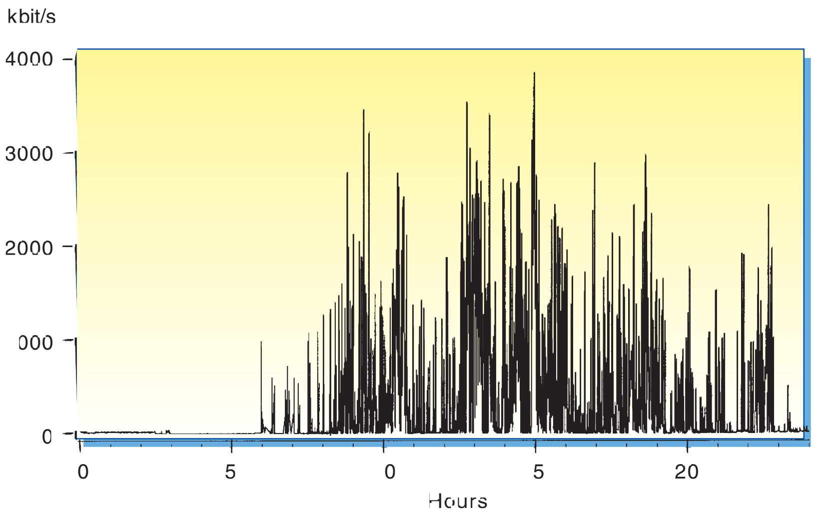


Figure 5.3 Outgoing traffic load during one day, category E

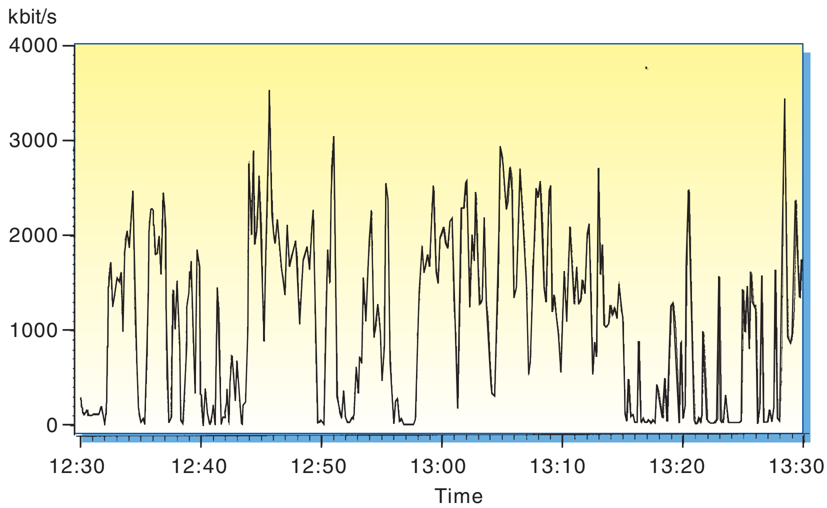


Figure 5.4 Outgoing traffic load during busy-hour, category E

outgoing traffic has a larger traffic volume than the incoming for this particular category E site. The asymmetric traffic load between the incoming and outgoing traffic load is visible in Figures 5.1 and 5.2.

Consequently, the outgoing traffic will be of interest in order to evaluate the symmetric WAN-link capacity. The one day traffic pattern for the traffic leaving this LAN site is given in Figure 5.3, and the busy-hour for that day is shown in Figure 5.4.

The outgoing traffic pattern contains several peaks exceeding the 1024 kbit/s limit, peaks up to 3.87 Mbit/s are detected. These peaks have a width of 10 seconds, as explained in Section 2.4, and will be buffered in the router until the WAN link is able to transport them. Stand-alone peaks 'taller' than the WAN capacity, one or two integration time periods of 10 seconds wide, will only experience short delays in the router before they are transmitted on the WAN link.

During the busy hour in Figure 5.4, found by inspection to be from 1230 to 1330 hours, the *average* traffic load applied to the router was 1077 kbit/s, as indicated in Table 5.2. Naturally, a traffic load from the LAN site this intense and continuing will saturate the WAN link, as the traffic load exceeding the WAN link capacity cannot be handled instantly by the router, but will have to be buffered up. Some of the generated traffic may stem from retransmissions due to buffer overflow in the router, causing time-outs in the end systems. Clearly, such long lasting saturation situation will cause severe delays for the users.

Table 5.2 provides statistical information about the traffic load for Figure 5.1 through 5.4, in terms of average and maximum bit/s per 10 seconds integration interval, and the total number of bytes. Two relative numbers are also included in the tables: 1) *coefficient of variation* (CoV), the ratio of the standard deviation to the mean, which indicates the relative variation of the data in the time window; and 2) *relative mean,* the ratio of the mean to the three days mean, that compares the mean of the data in actual time window to the overall three days mean.

The statistics for the one day time window show little difference from the three day window, except for the fact that the average outgoing traffic load is 13 %

higher for the actual day compared to the three day mean. These mean values are calculated for the whole time window duration. Considering Figure 5.3, it is visible that there is a 6 hour period during the night where there is hardly any traffic at all. Taking the total traffic load for the day and dividing it by the remaining 18 hours yields a mean for periods of the day with traffic equal to 443,508 bit/s, which is a more representative mean.

The busy-hour mean is 3.65 times larger than the three day mean. The CoV for the busy-hour is also smaller than for the other time windows, since the traffic load is on a more stable level during the busy-hour compared to the rest of the measurement period.

5.4 LAN interconnection traffic demands to WAN services

The characteristics of LAN interconnection traffic comprise three main requirements for existing and future WAN services: 1) asymmetrical transmission capacity, 2) efficient traffic mechanisms for bursts, and 3) flexible allocation of capacity.

The *asymmetrical* traffic load is due to the client-server traffic, where the server to client traffic can be up to 7 times the load in the opposite direction. Clearly, symmetrical WAN links, as today's leased lines, must be dimensioned with respect to the server to client traffic. This results in idle bandwidth in the less loaded traffic direction, which represents extra transmission costs for the customer and over-booking of network resources for the network operator.

Customers in the existing data packet services, like X.25 and Frame Relay, are charged on the basis of carried traffic volume, where a different traffic load per direction is taken into account. Also, the customer is charged for the symmetrical subscriber line into the nearest X.25 or Frame Relay node. The latter is a leased line with a fixed tariff, regardless of traffic volume. An early ATM service, with leased subscriber lines into the nearest ATM node, is expected to follow a similar philosophy. When ATM has been deployed to the customers' premises, a tariff totally dependent on asymmetric traffic loads may be possible.

The *bursty* nature of the LAN interconnection traffic represents high demands in the dynamics of the WAN services. Present synchronous leased lines services offer a constant capacity, which means that the WAN link must be dimensioned for the peak bursts in order to give all traffic a good quality of service. As seen in Table 5.1, the LAN interconnection is under-dimensioned with respect to the peak loads in order to save transmission costs, resulting in severe delays for the end users during extreme load situations.

The statistical multiplexing in X.25, Frame Relay and ATM will, to a certain extent, support transmission of random peak loads of short duration from one customer, provided that the link is not saturated with traffic from other customers. The ability to handle extreme peaks is dependent on the buffer sizes in the nodes. The traffic control mechanisms will take action if the network becomes overloaded. In order to prevent network congestion, the nodes can dispose of low priority packets/frames/cells or the network can refuse to accept any more incoming packets until the overload situation is relieved.

The *flexibility* requirement for WAN services is due to the alternating use of the LAN interconnection. A branch office with normal traffic load during the working hours, may transmit larger quantities of back-up traffic at off peak times, e.g. at night. In this case it will be feasible to change the service capacity in both traffic directions, either on an on-demand basis or according to a fixed schedule. This may be taken care of by future network or service management facilities, where the user, on-line or via the network operator, can change the service attributes according to instantaneous LAN interconnection demands.

6 Conclusions

The main conclusions from the LAN interconnection traffic measurements are:

- Branch offices do generally have larger incoming than outgoing traffic. This is due to relatively larger responses than requests from remote servers, usually situated in the head office.
- Branch offices without local servers have significantly larger external traffic than branch offices with servers on site. In the latter case, standard office applications are served locally, while servers for central applications and databases are situated in remote head offices.
- Head offices have larger outgoing than incoming traffic, which is due to the central server traffic explained above.
- CAD applications generate considerably more data traffic per user than standard office packages.
- IP counts for 60 % of the traffic to and from the selected LAN sites. The most used protocols on the application layer are Novell's NCP, Telnet over TCP/IP and IBMs NetBEUI.
- Generally, bulk transfer protocols, such as FTP and NFS, contains larger frames than interactive protocols, such as Telnet and Xwindows.
- The mean external traffic load during the busiest hour was under 90 % of the theoretical WAN link capacity for 11 out of 12 LAN sites. One LAN site generated outgoing traffic equivalent to 105 % of the WAN link capacity during the busy hour. The overload has to be buffered up in the router.

Table 5.2 Traffic load statistics for different time scales, category E

Time scale	Direction	Mean bit/s	Max. bit/s	Total number of bytes	1) CoV	2) Rel. mean	Figure ref.
3 days	Incoming	164 742	1 947 064	4 914 885 881	1.39	-	5.1
3 days	Outgoing	294 571	3 865 168	8 418 091 361	1.74	1.00	5.2
1 day	Outgoing	332 631	3 865 168	2 885 985 889	1.71	1.13	5.3
1 hour	Outgoing	1 076 560	3 528 670	484 452 053	0.81	3.65	5.4

1) *CoV – Coefficient of Variation, the ratio of the standard deviation to the mean. [4]*
2) *Relative mean – the mean for the time window relative to the 3 day mean.*

- The external LAN traffic is extremely bursty during high load situations, and peaks, of 10 seconds duration, were up to 7 times the available, theoretical WAN link capacity.

These observations yield requirements on the future and existing WAN services that shall carry the LAN interconnection traffic. The client-server traffic has proven *asymmetrical,* therefore, WAN links must supply bandwidth per direction. Also, the public network must be capable of handling *bursty* LAN traffic without severe delays or lost frames. Finally, services must be *flexible,* in order to offer required capacity according to instantaneous LAN interconnection traffic demands.

7 Suggestions for further work

A study of cost versus performance for the WAN services used for LAN interconnections is an appropriate next step for utilizing the results from measurements of the type presented in this report. The obtained external traffic measurement data for a particular LAN site can be applied to a suitable WAN service, and issues like tariff, throughput, delay and buffer sizes should be looked into.

In this work, the whole LAN has been regarded as a source and sink for external traffic, only recognized by its applications, configuration and number of users. In order to understand the applications generating the traffic on the LAN interconnection, a study per application should be performed. One station running a certain application, e.g. FTP, can be defined as a source and sink, and this specific traffic can be analysed with respect to traffic intensity and MAC frame characteristics. Then, a large number of stations running the same application could be employed, and the results compared to the one station example would give an indication of the traffic load per station for different number of stations.

Measurements with shorter integration times than 10 seconds for predefined periods of the day should be conducted for better reconstruction of extreme traffic peaks. As WAN protocol analysers become more advanced, the measurements can be carried out on the WAN side, which would give an exact picture of the LAN interconnection traffic flowing in the public network.

The integration time introduced in these measurements hides information about the exact traffic patterns for data frames, packets or cells. For modelling of LANs as ATM sources, measurements at the cell level are required. For higher level queue models, obtained distributions of mean values for frame interarrival times and frame sizes can be used. In this case, the interarrival time distribution of means will represent the *arrival* or *departure* rate, and the frame size distribution of means will represent the *service* rate.

8 References

1 Gaaren, S. *LAN interconnection traffic measurements : the results.* Kjeller, Telenor Research, 1994. (TF R 52/94.)

2 Gaaren, S. *LAN interconnection traffic measurements : the methodology.* Kjeller, Telenor Research, 1994. (TF R 51/94.)

3 Leland, W E et al. On the self-similar nature of ethernet traffic. I: *SIGCOMM '93,* New York, ACM, 1993.

4 Jain, R. *The art of computer systems performance analysis.* New York, Wiley, 1991. ISBN 0-471-50336-3.

5 Acharya, M K et al. A flow model for computer network traffic using real-time measurements. I: *2nd conference on telecommunication systems, modelling and analysis,* Nashville, Tennessee, 1994, 141–149.

6 Heegaard, P E, Helvik, B E. *ATM traffic experiments : source type models and traffic scenarios.* Trondheim, SINTEF DELAB, 1994.

7 Falk, G, Michel, T (eds.). *Interop 93 : LAN interconnection architectures : solutions, trade-offs, and trends.* San Francisco, CA, 1993.

8 Miller, M A. *Internetworking : a guide to network communications LAN to LAN : LAN to WAN.* Hemel Hempstead, Prentice Hall, 1991. ISBN 0-13-365974-7.

9 Caceres, R et al. Characteristics of wide-area TCP/IP conversations. I: *SIGCOMM '91,* New York, ACM 1991.

Aspects of dimensioning transmission resources in B-ISDN networks

BY INGE SVINNSET

1 Introduction

The complexity of current telecommunication networks makes it very important to have good network planning routines to meet the demands of the customers with the necessary investments. One important part of this is the dimensioning of the needed transmission resources, and a subtask is the dimensioning of the number of channels needed between each pair of network switching nodes. This is what we understand by dimensioning of trunk groups. Input to this task is the forecast traffic demands and the mapping of these demands onto the network routes where the various calls can be placed. This is then again mapped onto the trunk groups taking part in the route.

In telephony a channel is a dedicated physical resource that must be reserved for a single call for the whole lifetime of the call. In today's digital networks these physical resources are divided in time-slots (Time Division Multiplexing, TDM). Every Nth slot is reserved for a single call, i.e. every Nth slot belongs to one channel where N depends on the rate of the physical medium, e.g. 2 Mb/s or 34 Mb/s.

In ATM a channel has a more complex definition and the amount of resources needed for one channel is not static but depends on the needed data rate and the needed quality of service. The data rate of a channel may also vary in time. This complexity gives an increased flexibility and a possibility for saving capacity (statistical multiplexing gain) and these are also two of the reasons for the attractiveness of ATM. But it also creates new problems and new challenges for the network planning task.

In this paper we will present some methods for dimensioning of the required capacity of trunk groups in an ATM network. The task is to provide enough capacity for the forecast traffic needs and at the same time satisfy the given quality of service constraints. A subtask of this is to look for ways of dividing capacity between different traffic types to obtain a more effective utilisation of the network resources. The intention of such a division is to give a lower dimensioned capacity than we would have obtained without using such methods. This division will have implications for the connection admission procedure.

An ATM link (= transmission path in [5]) is in this context a physical transmission channel, characterised by a constant bit rate (for example 150 Mb/s or 600 Mb/s) and extending between a pair of transmission path connection end-points. A virtual channel connection (VCC) is an end-to-end virtual connection, set up between two users and characterised by traffic parameters (for example peak bit rate and mean bit rate). A virtual path connection (VPC) is set up between two ATM switching nodes, between a user and a switching node or between two users. It can be characterised in the same manner as a VCC. An ATM link can contain one or many VPCs and a VPC can contain one or many VCCs. For more exact definitions and more information about these concepts we refer to [5].

This paper considers only VPCs or VCCs, i.e. we do not treat the problems related to bundling VCCs into VPCs. Except for the first part of this paper where we treat the multi-dimensional Erlang loss formula, we only treat linear Connection Admission Control (CAC) functions. The connection admission is then based on the calculation of an equivalent bandwidth for each connection. Ignoring the Cell Delay Variation (CDV) tolerance this equivalent bandwidth may equal the peak bit rate or may be less than the peak bit rate but higher than the mean bit rate. Taking CDV tolerance into account the equivalent bandwidth may become higher than the declared peak bit rate if the CDV tolerance is allowed to take a high value.

2 Performance parameters that impact dimensioning

Performance parameters are divided into direct and derived performance parameters. Derived parameters such as availability will not be considered here.

Direct performance parameters can be divided into

- call/connection processing performance parameters
- information transfer performance parameters.

The connection processing performance parameters concern the establishment, release and parameter change of connections. These are

- connection rejection probability
- connection set-up delay
- connection release delay
- in-call connection parameter change delay.

The information transfer performance parameters to be considered are

- cell error ratio
- cell loss ratio
- cell misinsertion rate
- severely errored cell block ratio
- cell transfer delay
- cell delay variation.

Of these performance parameters the two parameters that have most impact on trunk dimensioning are:

- connection rejection (blocking) probability, and
- maximum cell loss ratio.

The connection blocking probability is the probability that the network is not able to deliver a connection between two users due to lack of network resources or due to errors in the network. A network dimensioned for every possible circumstance will be a very expensive network, if possible to realise at all. For this reason we need to set a positive value for this parameter, a connection blocking objective, taking both cost and user satisfaction into account.

The connection blocking objective may differ between the services. If no service protection is used the high capacity services will certainly experience a higher probability of connection blocking than the low capacity services. This can be very unsatisfactory. Although it is not necessary to offer the same maximum connection blocking probability for all services, it is necessary to use some kind of service protection method to control the quality of service. Service protection methods should be used to dimension the network with specified connection blocking objectives for the different services. Dimensioning should be made for busy hour in the same manner as for traditional circuit switched networks, but the busy hour may be different for the different services.

When it comes to the information transfer performance, different services will have different requirements with regard to cell delay, cell delay variation and cell loss ratio. For the exact definitions of the information transfer performance parameters, see [6]. Some services like telephony are not so sensitive to cell losses, but cell delay and delay variation must be small. For data services the requirement for maximum cell delay is not so strict, but the requirement for maximum cell loss ratio is stronger than for telephony. To cope with such problem QoS classes (performance parameter classes) on the cell level could be introduced. Otherwise the network will have to be dimensioned for the strongest requirements.

Some data services require large buffers. A service like telephony which is delay sensitive requires small buffers. Some kind of buffer management scheme will have to be implemented, possibly combined with the use of separate physical links in the network, to satisfy the needs of both these services in the same network.

For a given buffer size the maximum link allocation that the CAC can permit is given by the cell loss requirements. This maximum link allocation determines the capacity of the physical link in terms of the amount of bandwidth that can be allocated to the virtual connections crossing that physical interface. This then has direct implication on the number of physical interfaces needed towards a specific destination (i.e. for the trunk group) to satisfy the bandwidth needs determined by the trunk group dimensioning.

3 The use of QoS classes

QoS classes may be introduced to cope with different requirements for maximum cell loss ratio and cell delay variation, and may improve the utilisation of the network depending on the traffic mix. A way to achieve this is by use of the Cell Loss Priority (CLP) bit in the ATM header and some kind of buffer administration that gives priorities to cells depending on the value of the CLP bit. In this way we will have two QoS classes, for example with cell loss requirements of 10^{-9} for the class with highest priority and 10^{-7} for the class with the lowest priority.

The priority bit can be introduced at the cell level so that cells belonging to the same connection can have different priorities. This has been suggested for video services, but it seems not so much discussed in standardisation bodies anymore.

The priority bit can also be used at the connection level. This leads to a classification of connections into two QoS classes depending on their requirement for maximum cell loss ratio. It is still unclear how to use the CLP bit, and the first switched ATM networks will probably use other methods to distinguish between QoS classes because it is possible to offer more than two classes by using other means than the CLP bit. One way is to use for instance the VPI field combined with some kind of buffer administration. Another way is to use some signalling or management message in the set-up phase to be able to allocate the connections to dedicated ATM links for the different QoS classes or to enable buffer management procedures to discriminate between the different connections.

Different ways of dividing connections into QoS classes have been proposed, and it seems that at least three such classes will be implemented in early switched ATM networks:

- A QoS class with a requirement for low cell loss (e.g. max 10^{-9}) and high CDV tolerance or no end-to-end requirement on CDV (data applications).
- A QoS class with a more relaxed requirement for cell loss (e.g. max 10^{-6}) but with a low CDV tolerance (voice and interactive video).
- A QoS class with unspecified values ('best effort' and 'low cost' services).

The use of different links for the different classes is only of interest if the offered traffic is high enough to justify the reservation of ATM links for the different classes. Anyhow, QoS classes with different requirements for maximum cell loss will have impact on dimensioning dependent on the way it is implemented. If separate ATM links and buffers are used with possible common overflow links, we can dimension the amount of resources needed for each QoS class separately. In this case each connection belongs to one QoS class, and the requirement for maximum cell loss ratio has implications for the maximum utilisation of the ATM link, that is the maximum number of simultaneous connections on each link. The overflow links will have to be dimensioned for the class with the strongest requirements. If buffer administration is used, this implies that connections belonging to different QoS classes may share the same ATM link but experience different cell loss ratios. Calculation of the admissible loading of the links will then be more complicated and is a matter for future research.

4 Dimensioning of trunk groups in a complete sharing environment

The task of this chapter is the dimensioning of the transmission resources in terms of the number of ATM links needed or the total bandwidth needed between two network nodes. The set of ATM links between two nodes is termed a trunk group.

In this chapter we will concentrate on the general methodology for the dimensioning of trunk groups. This methodology is based on mathematical models that approximate the statistical behaviour of the traffic in the network. Inputs to this problem are the services specified as traffic types with certain traffic parameters, the offered traffic and a connection blocking objective for each traffic type. We will be faced with the difficulty of having services with different connection blocking objectives or we have to provide some service protection method (or both). This problem will be treated in the next chapter. We may also be faced with the problem of having traffic components with different QoS objectives even on the ATM level (i.e. maximum cell loss ratio). This problem was discussed in chapter 2 and will not be elaborated any further.

The connection blocking objective is the *connection blocking at the connection level* for a trunk group, defined as the proportion of the offered connections that are blocked. The connection is blocked if the connection is not accepted by the CAC function for any of the ATM links in the trunk group. All blocked connec-

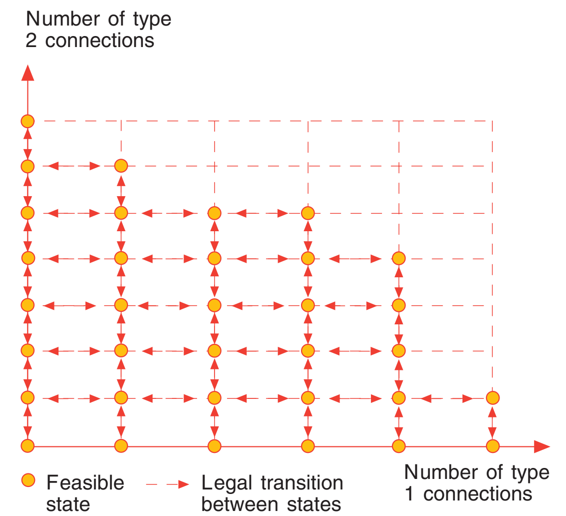


Figure 1 Reversible transition diagram and coordinate convex set of feasible states for an ATM link with 2 traffic types

tions are cleared. In a multiservice network such as an ATM network, the probability of blocking is dependent on the offered traffic mix and on the traffic type.

We should normally take as a starting point the end-to-end connection blocking (rejection) probability for each traffic type. This is given as a part of the performance specification of the network. With basis in a network reference configuration, these performance measures can be decomposed and allocated to trunk groups in the network. In the following we therefore assume that we have given a maximum connection blocking probability for each traffic type offered to a given trunk group. In addition to this, for a real dimensioning, we need forecasts of the traffic for each traffic type for a reference period (busy hour), and we will assume that the traffic process is stationary during the reference period. We will also assume that all traffic streams are independent.

4.1 The multi dimensional Erlang loss formula

Let K be the number of traffic types offered to a trunk group. The CAC may now generally be formulated in the following way:

> n_{1v} connections of traffic type 1, n_{2v} connections of traffic type 2, ..., n_{Kv} connections of traffic type K may at the same time be acceptable on the ATM link v if the CAC function
>
> $F(n_{1v},n_{2v},...,n_{Kv}) \leq C$
>
> where C is the capacity of the ATM link (i.e. *the maximum permissible capacity allocation*).

So, $(n_{1v},n_{2v},...,n_{Kv})$ is a *feasible state* (allowable state) for this ATM link if and only if this is the case. If $(n_{1v},n_{2v},...,n_{Kv})$ is a feasible state for ATM link v, $v = 1,...,V$ where V is the number of ATM links in the trunk group, then $(n_{11},n_{21},...,n_{K1},...,n_{1V},n_{2V},...,n_{KV})$ is a feasible state for the trunk group. We denote by Φ the set of all feasible states for the trunk group. Φ is also called the acceptance region for the set of occupancy states.

A trunk group consisting of more than one ATM link is not a full availability group in the sense that a connection can not be divided between links. That is, it is not enough to know the total number of connections of each type in the system. We have to know the distribution of the connections among the ATM links in the group. Blocking performance is then dependent on the hunting strategies. A random hunting strategy within the group will place connections on ATM links with available capacity in random order. This may give a bad blocking performance for high capacity connections because spare capacity is fragmented between the links. The same is true for sequential hunting at variable position. A connection packing strategy that seeks to place connections on the most heavily loaded available ATM link gives a better blocking performance for high capacity connections. If we then can assume, as an approximation, that the trunk group is a full availability group (complete sharing), the state of the trunk group can be written $(n_1,n_2,...,n_K)$ where

$$n_i = \sum_{v=1}^{V} n_{iv}$$

The dimensioning problem is then reduced to the problem of calculating the total bandwidth needed on the ATM level for this trunk group. The mapping of this bandwidth onto physical transmission systems, taking into account that different transmission systems with different bandwidths may be used in series and in parallel, is not considered here. Transmission overhead at the physical layer is not considered either.

Now assume that the arrival process for the traffic is either a Bernoulli process, a Poisson process or a Pascal process. That is, the arrival intensity for traffic type i is $\lambda_i(n_i)$ where n_i is the total number of connections of type i in the system and where:

- $\lambda_i(n_i) = \max\,[0,(N_i - n_i)\gamma_i]$ for a Bernoulli process,
- $\lambda_i(n_i) = \lambda_i$ for a Poisson process, and
- $\lambda_i(n_i) = \alpha_i + \beta_i \cdot n_i$ for a Pascal process.

The Bernoulli process corresponds to a random process with N_i sources for traffic type i. Each of these sources then, when they are not active, tries to connect with an intensity γ_i. These are forecast intensities. When the number of sources is high $(N_i \gg n_i)$ we use the Poisson process. The Pascal process is normally used for overflow traffic.

We further assume that the mean value of the holding time is $h_i = 1/\mu_i$ for traffic type i. If the CAC function F is strictly monotone in $(n_1,n_2,...,n_K)$, then the state transition diagram is reversible and the set Θ of feasible states is coordinate convex. This means that if $(n_1,n_2,...,n_i + 1,...,n_K)$ is a feasible state, then $(n_1,n_2,...,n_i,...,n_K)$ is also a feasible state, and then state $(n_1,n_2,...,n_i + 1,...,n_K)$ can be reached by an arrival of a type i connection when the trunk group is in state $(n_1,n_2,...,n_i,...,n_K)$ and the state $(n_1,n_2,...,n_i,...,n_K)$ can be reached by a termination of a type i connection when the trunk group is in state $(n_1,n_2,...,n_i + 1,...,n_K)$.

In this case it can be shown (see [1] for Bernoulli/Poisson processes and [10] for Pascal processes) that the state probabilities have a product form solution. That is:

$$p(n_1,n_2,\ldots,n_K) = c^{-1}\cdot\prod_{i=1}^{K}\left(\frac{\prod_{j=0}^{n_i-1}A_i(j)}{n_i!}\right)$$

for a feasible state $(n_1,n_2,...,n_K)$ (1)

where

$$c = \frac{1}{p(0,0,\ldots,0)} = \sum_{(n_1,n_2,\ldots,n_K)\in\Phi}\left(\prod_{i=1}^{K}\left(\frac{\prod_{j=0}^{n_i-1}A_i(j)}{n_i!}\right)\right)$$

is the normalisation constant and

$$A_i(j) = \frac{\lambda_i(j)}{\mu_i}$$

is the offered traffic of traffic type i when j connections of type i are connected to the system.

This result is valid irrespective of the holding time distributions of the traffic types [10], so every traffic type may have its individual holding time distribution.

$(n_1,n_2,...,n_K)$ is a *blocking state* for traffic type i if $(n_1,n_2,...,n_i,...,n_K)$ is a feasible state and $(n_1,n_2,...,n_i+1,...,n_K)$ is a *non-feasible* state. We denote by B_i the set of all blocking states for traffic type i. This set is also called the blocking boundary. The concept of blocking states is illustrated in Figure 2 for a case with 2 traffic types.

The time blocking for traffic type i is the proportion of the time the system is blocked for a new connection of type i.

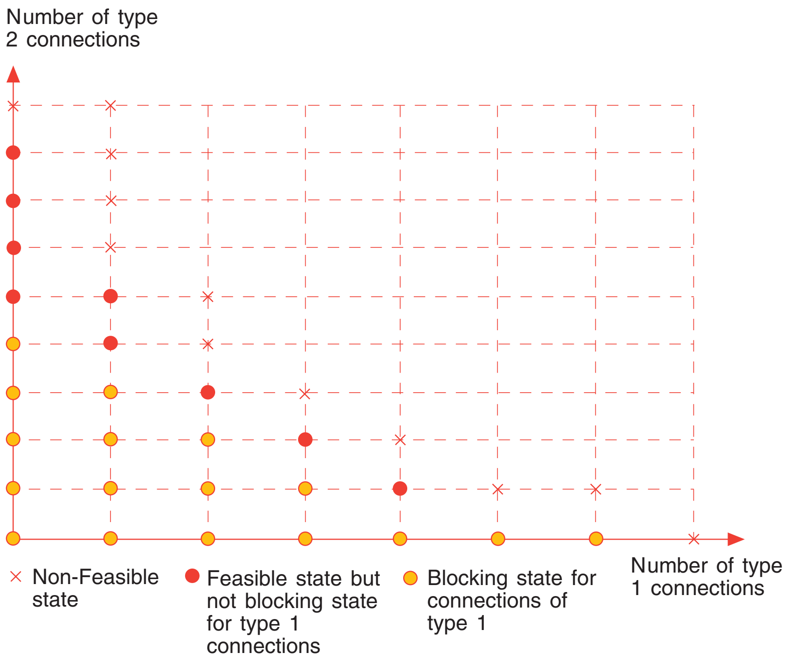


Figure 2 The concept of blocking states for a connection type. Only blocking states for connections of type 1 is indicated

This is the same as the probability that the system is in a blocking state for traffic type i, i.e.:

$$E_i = \sum_{(n_1,n_2,\ldots,n_K)\in B_i} p(n_1,n_2,\ldots,n_K)$$

It can be shown that the relation between connection blocking and time blocking is given by:

- $B_i(N_i,\gamma_i) = E_i(N_i - 1,\gamma_i)$ for a Bernoulli process
- $B_i(\lambda_i) = E_i(\lambda_i)$ for a Poisson process, and
- $B_i(\alpha_i,\beta_i) = E_i(\alpha_i + \beta_i,\beta_i)$ for a Pascal process.

On the basis of the forecast traffic for each traffic type between two network nodes, we can now use this general model to calculate the total bandwidth needed for a trunk group to satisfy the connection blocking objective for each traffic type. For direct trunk groups without overflow traffic we will assume that the arrival process is a *Poisson process* so that the total offered traffic is independent of the system state and the connection blocking and the time blocking is equal. The formula can then be simplified, but when K increases the formula is not so tractable anyhow because of all the states involved. Especially we have the problem of calculating the normalisation constant. Various methods have been developed to tackle this problem. They are generally based on recursions of the Buzen type on the state probabilities. However, the state description implied in the multi-dimensional Erlang formula is too detailed and prevents a fast evaluation of the blocking probabilities.

In order to compute more efficiently the blocking probabilities, Kaufman [9] and Roberts [12] introduced an aggregate state description for the number of occupied channels (i.e. allocated bandwidth). The one-dimensional recursive algorithm they derived will be treated in the next section and is based on the assumption of a linear capacity allocation (which in case of ATM systems has to be performed by the CAC).

More precisely, each traffic type i is characterised by an effective bandwidth c_i which is supposed to be occupied by each connection of type i on the link. In the case of peak rate allocation, c_i is simply the peak rate of the connection. In the case of Variable Bit Rate services c_i is a value between the mean rate and the peak rate. If the CDV tolerance is high, the value may become higher than the peak rate.

4.2 Survey of models for dimensioning trunk groups in case of linear CAC

We will in the following treat the trunk group as consisting of only one ATM link whose capacity it is our task to dimension and we will only treat CAC functions based on a constant effective bandwidth for each traffic type. This means that with effective bandwidth c_i for traffic type i and the ATM link in state $(n_1,n_2,\ldots,n_K)$, the condition for accepting a new connection of type i is that

$$\sum_{j=1}^{K} c_j \cdot n_j \le C - c_i$$

where C is the capacity of the ATM link (i.e. the maximum permissible capacity allocation). In the next chapter we will extend this model with service protection methods.

Let us now assume that the values $c_1,\ldots,c_K$ have a greatest common divisor $\Delta c = gcd\{c_1,\ldots,c_K\}$.

This can then be viewed as a basic bandwidth unit and methods developed for multislot networks can be applied. Let further

$$d_i = \frac{c_i}{\Delta c} \text{ and } D = \left\lfloor \frac{C}{\Delta c} \right\rfloor .$$

4.2.1 Roberts and Kaufmann recursion formula

We define the overall occupancy distribution as

$$Q(d) = \sum_{\Sigma n_i \cdot d_i = d} p(n_1,\ldots,n_K)$$

The blocking probability for traffic type i is then

$$E_i = \sum_{d > D - d_i} Q(d)$$

The following recursion algorithm for $Q(d)$ was first presented in [12] and [9]:

$$d \cdot Q(d) = \sum_{i=1}^{K} A_i \cdot d_i \cdot Q(d - d_i)$$

for $d \le D$ where $Q(d\text{-}d_i) = 0$ if $d < d_i$.

This recursion algorithm is also valid if the effective bandwidths have no greatest common divisor and the bandwidth demands are given in Mb/s instead of basic bandwidth units. But the computational effort will then be the same as for the general model and we have gained nothing. The computational effort will be much less if a basic bandwidth unit can be found.

This recursion formula has been extended to Bernoulli and Pascal arrivals in [2], but in these cases we only get approximate values for the connection blocking probabilities.

4.2.2 Convolution algorithm

This algorithm for calculating the global state probabilities (1), is recursive in the number of sources. The algorithm was first published in [7]. In the paper it is shown that it is allowed to truncate the state space at K (K = the number of traffic types) and renormalise the steady-state probabilities. For Poisson traffic this method is not so fast as the recursion formula above. For Bernoulli and Pascal traffic this method gives exact values, not only for the time blocking probabilities, but also for the connection blocking probabilities.

The algorithm goes in three steps. In the first step we calculate the state probabilities $p_i(n_i)$ ($n_i = 0, ..., n_i^{max} = D/d_i$) for each traffic type i as if this traffic type was alone in the system. For this we can use the product form expression (1) with $K = 1$.

In the next step we calculate the global state probabilities by successive convolutions:

$Q_1(n_1 \cdot d_1) = p_1(n_1)$ for $n_1 = 0,...,n_1^{max}$

$$Q_i(d) = \sum_{j=0}^{d/d_i} Q_{i-1}(d - j \cdot d_i) \cdot p_i(j)$$

for $d = 0,...,D$ and $i = 2,...,K$

The global state probabilities are now given by $Q(d) = Q_K(d)$ for $d = 0,...,D$ after normalising.

The time blocking probability can then be calculated as above, but for Bernoulli and Pascal arrivals we need another step to calculate the connection blocking probabilities.

In this third step we deconvolute $Q(d)$ by finding $Q^i(d)$ such that

$$Q(d) = \sum_{j=0}^{d/d_i} Q^i(d - j \cdot d_i) \cdot p_i(j)$$

The connection blocking probability for traffic type i is now given by:

$$B_i = \frac{\sum_{d=D-d_i+1}^{D} \sum_{j=0}^{d/d_i} \lambda_i(j) \cdot Q^i(d - j \cdot d_i) \cdot p_i(j)}{\sum_{d=0}^{D} \sum_{j=0}^{d/d_i} \lambda_i(j) \cdot Q^i(d - j \cdot d_i) \cdot p_i(j)}$$

Due to numerical problems it may be better to calculate Q^i by convolutions of homogenous state probabilities p_i.

4.2.3 Lindberger method

This is an approximate method.

Let $d_{max} = \max\{d_1,...,d_K\}$ and $n = D - d_{max}$. We assume that d_{max} is much smaller than n. The blocking states are now approximated by [11]:

$$Q(n+k) \approx \frac{\mathrm{Erl}\left(n', \frac{A}{z}\right)}{z} \cdot \left(\frac{A}{D}\right)^{\frac{k}{z}}$$

for $k = 1,...,d_{max}$, where

$$A = \sum_{i=1}^{K} A_i \cdot d_i$$

is the total bandwidth demand of the offered traffic,

$$z = \frac{\sum_{i=1}^{K} A_i \cdot d_i^2}{A}$$

is the peakedness factor (and a measure of the mean number of slots for a connection in the mixture ('equivalent average call'), see [3]),

$$n' = \frac{n + \frac{1}{2}}{z} - \frac{1}{2}$$

and Erl(x,A') is the Erlang formula (linear interpolation between integer values of x).

The validity of this method for ATM should be tested and perhaps other values of n' should be chosen depending on the traffic mix.

Now the idea is to divide the traffic types into two classes and to dimension capacity using the model above for each of these classes separately. The first class is for low capacity traffic, i.e. suggested for connections with a highest peak rate / effective bandwidth in the order of 0.5 Mb/s on a 155 Mb/s ATM link. For this class the objective is to control the average connection blocking probability

$$E = \frac{\sum_{i=1}^{K} A_i \cdot d_i \cdot E_i}{A}$$

(only the low capacity traffic types are considered now), and the highest individual connection blocking probability in the mixture $E_{max} = \max\{E_i \,|\, i = 1,...,K\}$.

If $d_{min} = \min\{d_1,...,d_K\}$ approximate formulas for these are

$$E \approx \mathrm{Erl}\left(\frac{D - z + d_{min}}{z}, \ \frac{A}{z}\right)$$

$$E_{max} \approx \frac{d_{max}}{z} \cdot \left(\frac{D}{A}\right)^{(d_{max}-z)/2z} \cdot E$$

For the other class, i.e. suggested for connections with peak rate/effective bandwidth between 0.5 Mb/s and 5–10 Mb/s, the requirement for connection blocking probability is not so strong. For this class it is suggested to use the same connection blocking requirement for all traffic types in the class (trunk reservation, see below). An approximation for this blocking probability is given by

$$E_0 \approx \mathrm{Erl}\left(\frac{D - d_{max} + d_{min}}{z}, \ \frac{A}{z}\right)$$

Traffic types with higher capacity requirements will have to be treated separately, perhaps on a prebooking basis.

4.2.4 Labourdette & Hart method

This is another approximate method.

Let

$$A = \sum_{i=1}^{K} A_i \cdot d_i, \ V = \sum_{i=1}^{K} A_i \cdot d_i^2, \ \alpha,$$

the unique positive real root of the polynomial equation

$$\sum_{i=1}^{K} d_i \cdot A_i \cdot z^{d_i} = D$$

and

$$\zeta = \frac{\log\left(\frac{D}{A}\right)}{\log(\alpha)}$$

for $\alpha \neq 1$. The blocking probability for traffic type i is then by this approximation method given by [10]:

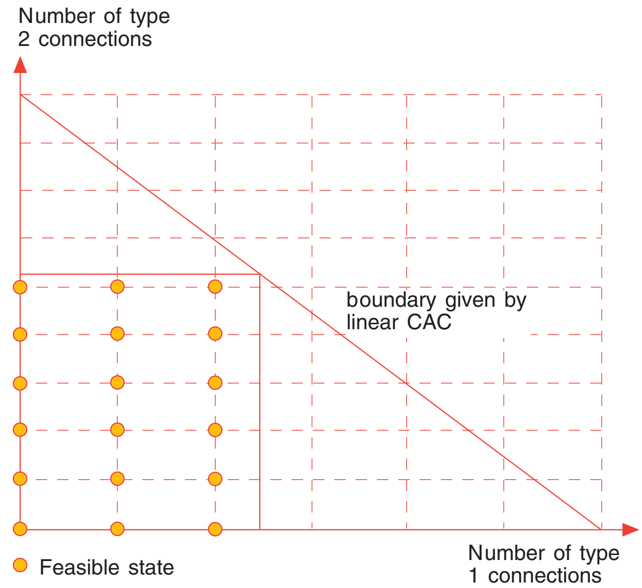


Figure 3 Set of feasible states with a completely partitioned trunk group, case with 2 traffic types

$$E_i \approx \begin{cases} \frac{1-\alpha^{d_i}}{1-\frac{D}{A}} \cdot \mathrm{Erl}\left(\frac{D}{\zeta}, \frac{A}{\zeta}\right), & \alpha \neq 1 \\ \frac{d_i \cdot A}{V} \cdot \mathrm{Erl}\left(\frac{D \cdot A}{V}, \frac{A^2}{V}\right), & \alpha = 1 \end{cases}$$

The Lindberger and Labourdette & Hart methods are meant for network planning purposes where the calculations must be iterated a great number of times in the process of optimising a network. For the purpose of dimensioning of the trunk groups between a network node and the adjacent network nodes the exact methods can be used.

5 Service protection methods

A complete sharing strategy will probably not be an optimal dimensioning strategy, though this depends on the traffic mix and the required maximum connection blocking probability for the different traffic types. The reason for this is that a complete sharing strategy will result in a higher connection blocking probability the higher the bandwidth demands are. An effect of this will be that requests for high bandwidth services will be rejected in heavy traffic periods due to lack of resources. Under normal conditions the connection blocking probability may be satisfactory, but then at the expense of a low network utilisation.

For trunk groups consisting of more than one ATM link, this may partly be remedied by using routing methods when hunting for free capacity. Two such routing methods are

- the concentration method, and
- the separation method.

The concentration method is a sequential hunting with homing. Since we then always start the hunting with the same link, this link will always be heavily loaded and the last link in the sequence will be less loaded (depending on the offered traffic).

When using the separation method we still do sequential hunting with homing, but the traffic types are divided in two classes (the types with the highest bit rates is one class) with opposite homing positions.

The advantage of these methods is a lower connection blocking probability for traffic types with high capacity demands. A disadvantage is a higher processor load due to more retrials before success in the hunting process.

Routing methods alone are not the best way to compensate for the bad effects of a complete sharing stategy and we have to use service protection methods for best network utilisation with given connection blocking constraints. Methods for protecting certain services in the connection establishment phase will normally give a lower dimensioned capacity for carrying the offered traffic streams with the constraints given by these connection blocking objectives. Such methods are described below.

5.1 Completely partitioned trunk groups

This is the straightforward service protection method in which case the trunk group is divided in K parts, one for each traffic type. It is illustrated in Figure 3 for $K = 2$. The needed capacity is dimensioned for each traffic type separately, based on offered traffic and connection blocking objective for this traffic type. Standard dimensioning methods can be used as if only this traffic type was offered to the system. The total dimensioned capacity will be the sum of the dimensioned capacity for each traffic type.

This method may be used in combination with one of the methods given below, i.e. the traffic types may be divided in classes with complete partitioning of the trunk group between the classes and use of one of the methods below inside a class.

5.2 Traffic type limitation method

This method is also called the partial sharing method.

The total number of connections of traffic type i is limited by

$$n_i \leq n_i^{\max} \leq \frac{D}{d_i}$$

for $i = 1,\ldots,K$, where

$$\sum_{i=1}^{K} d_i \cdot n_i^{\max} > D.$$

The method is illustrated in Figure 4 for a case with 2 traffic types,

$$n_1^{\max} = \frac{D}{d_1}, n_2^{\max} < \frac{D}{d_2}.$$

For this method we still get a product form solution for the state probabilities and we may also apply a modified version of Roberts and Kaufmanns recursion algorithm for the overall occupancy distribution:

$d \cdot Q(d) =$

$$\sum_{i=1}^{K} A_i \cdot d_i \cdot \left(Q\left(d-d_i\right) - L_i\left(n_i^{\max}, d-d_i\right)\right)$$

for $d \le D$.

Here $Q(d - d_i) = 0$ if $d < d_i$ and $L_i(n_i^{\max}, d - d_i)$ is the probability that $n_i^{\max}$ connections of type i are connected to the system and that $d - d_i$ slots are occupied in the system at the same time.

It is better to use the Iversen convolution algorithm in this case. The two first steps of this algorithm are then as before, only with the modification of $n_i^{\max}$ in the first step. We then need deconvolutions in the third step to calculate the blocking probability for connections of type i. That is, we find $Q^i(d)$ by:

$$Q(d) = \sum_{j=0}^{d/d_i} Q^i(d - j \cdot d_i) \cdot p_i(j)$$

The blocking probability is then given by

$$E_i = \sum_{d=D-d_i+1}^{D} Q(d) +$$

$$\sum_{d=0}^{D-d_i} Q^i\left(d - n_i^{\max} \cdot d_i\right) \cdot p_i\left(n_i^{\max}\right)$$

Methods for calculating the minimum total trunk group capacity to achieve the connection blocking criteria are given in [8].

5.3 Trunk reservation method

This method is also called the sum limitation method and the priority reservation method. It is probably more effective as a method for reducing the blocking probability for high capacity connections than the traffic type limitation method.

In this case we introduce a traffic type threshold to protect high capacity connections against high connection blocking probability. This means that with equivalent bandwidth $c_i = d_i \cdot \Delta c$ and the system in state $(n_1, n_2, ..., n_K)$, the condition for accepting a new connection of type i is that

$$\sum_{j=1}^{K} d_j \cdot n_j \le \Theta_i$$

with $\Theta_i \le D - d_i$ for $i = 1,...,K$, $\Theta_i < D - d_i$ at least for one value of i.

Here Θ_i is the link allocation threshold for traffic type i. The product form solution is not valid in this case. Good approximate solution can be found by the following recursion [13]: Approximate $Q(d)$ by the function $Q'(d)$ given by

$$d \cdot Q'(d) = \sum_{i=1}^{K} A_i \cdot D_i(d) \cdot Q'(d - d_i)$$

where

$$D_i(d) = \begin{cases} 0, d > \Theta_i + d_i \\ d_i, d \le \Theta_i + d_i \end{cases}$$

and

$$\Sigma Q'(d) = 1.$$

In [4] it is recommended, due to experience from simulation studies, to replace this expression by

$$d \cdot Q'(d) = \sum_{i=1}^{K} \frac{\lambda_i}{\mu} \cdot D_i(d) \cdot Q'(d - d_i)$$

where

$$\frac{1}{\mu} \cdot \sum_{i=1}^{K} \lambda_i \cdot (1 - E_i) \cdot d_i =$$

$$\sum_{i=1}^{K} \frac{\lambda_i}{\mu_i} \cdot (1 - E_i) \cdot d_i.$$

E_i is as before the connection blocking probability for traffic type i and the computation of the values E_i and μ along with the recursion algorithm must be iterated until the values converge.

By letting $\Theta_i = D - \max\{d_1,...,d_K\}$ the trunk reservation method gives equal blocking probability for all traffic types. More generally, one can tune the parameters Θ_i to obtain preferred connection blocking probabilities for the individual traffic types. A method for doing this is described in [4]. A problem with this method is instability which may arise in overload situations. To overcome this it may be that the link allocation thresholds Θ_i, as used by the CAC, must be dynamically changed in such situations.

6 Conclusions

We have discussed an extension of the Erlang loss formula to an ATM network with a finite number of connection types with different capacity demands and with a general connection acceptance procedure. When we limit the treatment to cases where the connection acceptance function is a linear function of the number of connections of the different connection types, fast algorithms exist for calculating the connection blocking probabilities. Thus, dimensioning of trunk groups based on objectives for the connection blocking probabilities, can be achieved rather straightforward in these cases. Linear connection acceptance functions will be dominating in early ATM networks.

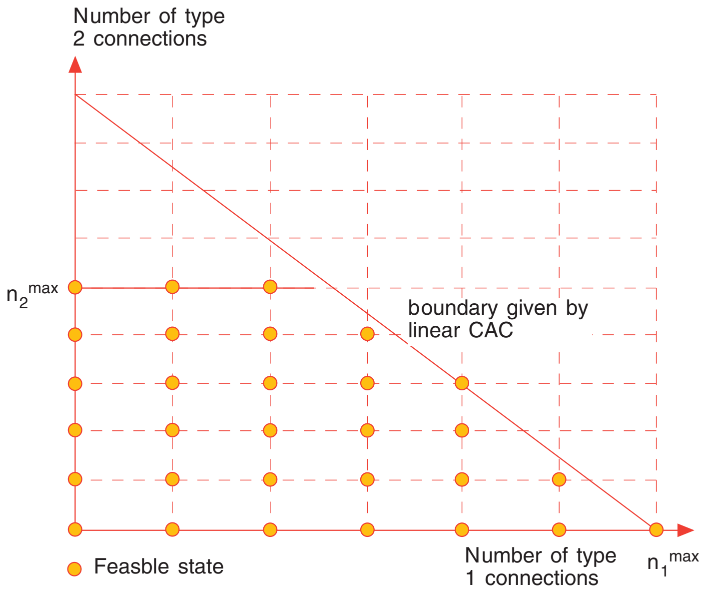


Figure 4 Traffic type limitation method

Preliminary studies indicate that traffic mixes where low capacity traffic competes with high capacity traffic, do not perform well. Either the connection blocking will be unacceptably high for the high capacity traffic or the link utilisation will be unacceptably low. Routing methods alone do not repair this, and we have to use service protection methods for best network utilisation with given grade of service constraints. Also in these cases the computational effort is small when dimensioning according to an equivalent bandwidth allocation model.

When the offered traffic grows, it may be a good idea to separate the traffic in classes with dedicated capacity pools in the trunk group for each class and perhaps a capacity pool for common overflow.

- Traffic with capacity demands of 5 – 10 % of the link capacity and more may be treated separately, perhaps on a reservation basis.
- It may be that we should also divide the rest of the traffic in two classes, i.e. traffic with capacity demands lower than for instance 0.5 % of the link capacity in one class, though this depends on the traffic volume and the traffic mix.

The provision of quality of service classes, best effort services, variable bit rate services with more complicated (non linear) allocation schemes and various ways of bundling virtual channel connections into virtual path connections, will complicate the dimensioning task. This is a challenge for future research activities.

References

1 Aein, J M. A multi-user-class, blocked-calls-cleared, demand access model. *IEEE transaction on communications,* Com 26, 3, 1978.

2 Delbrouck, L E N. On the steady-state distribution in a service facility carrying mixtures of traffic with different peakedness factor and capacity requirements. *IEEE transaction on communications,* Com 31, 11, 1983.

3 Fiche, G, Le Gall, P, Ricupero, S. Study of blocking for multislot connections in digital link systems. I: *11th international teletraffic congress, ITC-11.*

4 Gehrst & Lee. *Virtual-circuit load control in fast packet-switched broadband networks.* Globecom 1988.

5 ITU-T Recommendation I.311. *B-ISDN general network aspects.* Geneva 1993.

6 ITU-T Recommendation I.356. *B-ISDN ATM layer cell transfer performance.* Geneva 1994.

7 Iversen, V B. A simple convolution algorithm for the exact evaluation of multi-service loss systems with heterogeneous traffic flows and access control. I: *Proceedings of 7th Nordic teletraffic seminar, NTS-7.* Lund, Sweden, 1987.

8 Kaspher, A N. Bandwidth allocation and network dimensioning for international multiservice networks. I: *Proceedings of 5th international teletraffic congress, ITC-5,* Lake Como, 1987.

9 Kaufmann, J S. Blocking in a shared resource environment. *IEEE transaction on communications,* Com 29, 10, 1981.

10 Labourdette & Hart. Blocking probabilities in multitraffic loss systems: insensitivity, asymptotic behaviour and approximations. *IEEE transactions on communications,* 8, 1992.

11 Lindberger, K. Some ideas on GOS and call scale link-by-link dimensioning. I: *10th Nordic teletraffic seminar, NTS-10,* Århus 1992.

12 Roberts, J. A service system with heterogeneous user requirements. *Performance of data communications systems and their applications,* North Holland, 1981.

13 Roberts, J W. Teletraffic models for Telecom 1 integrated services network. I: *10th international teletraffic congress, ITC-10,* Montreal 1983.

Testing ATM switches

BY SVEINUNG O GROVEN

Abstract

The telecommunications industry is presently witnessing the broadband integrated services digital network (B-ISDN) evolution. B-ISDN, and specifically asynchronous transfer mode (ATM) technology as the chosen solution by ITU-T, may be looked back upon in a few years as being elementary and simple. Today this is definitely not the case. Experience in traffic profiles in the various public B-ISDN network topologies is limited. With various equipment and private networks connected to the public network the complexity will increase many times. The question asked by many is what traffic profiles will the future telecommunications networks experience. Some of these foreseen traffical scenarios and the influence these will have on traffic profiles are presented.

B-ISDN pilot projects are being carried out world-wide. Benchmark tests and performance evaluations are constantly being made. However likely or unlikely, various traffic models are suggested used in these tests. Thoughts around the validity of these tests are brought forward.

In order to test the Ericsson ATM switch, traffical switch performance tests were carried out. The theory behind these tests are outlined. Finally, a comparison of theory and reality based on practical test experience is made.

1 Introduction

One of the main problems associated with the dimensioning and testing of ATM networks is the fact that the standards are not yet completed. Experience gathered through tests will aid the completion of these standards, and that is why these performance measurements, and many other quality of service (QoS) measurements, are so important. Of course, before performance measurements can be made, conformance and interoperability testing must be completed. Even these tests reveal inconsistencies and misunderstandings of the standards and equipment being developed.

Figure 1 shows an example of a simple B-ISDN network. Even with such a simple network configuration the profiles of many different traffic sources are difficult to predict. Bearing in mind that this is an early configuration, measurements made today and the traffic profiles experienced at the measurement points of interest will be altered as the number of users, the amount of traffic, and the types of applications evolve.

1.1 A simple network scenario

Figure 1 gives a simplified view of a possible network scenario and some of the measurement points of interest.

Network topologies are one of the main factors which will influence the traffic profiles seen at the measurement points of interest. For example, whether a user network interface (UNI) is connected to a local area network (LAN) or is used for a video on demand (VoD) distribution service is of a major consequence to the traffic profiles.

Standardisation bodies estimate the performance objectives which should be provided by the network, but the end user and applications being run will determine the quality required.

The main objective of the tests outlined in this article was to estimate the performance of the switch under test. Some tests needed to be repeated many times, requiring quick and simple test routines, while others were simplified due to limitations in the test equipment.

1.2 Broadband services

Based on a basic knowledge of the wide range and variety of broadband services and applications being defined today, we realise that future applications may only be limited by our imagination. These may exhibit totally different traffic characteristics from those foreseen today. Bearing this in mind, we also realise the need to define reference points and standard test suites for performing QoS evaluations. In due time, experience will supply the information needed to dimension these networks properly.

1.3 ATM traffic

Various different traffic types have been specified to date. We have ATM traffic in its most simple form, Constant Bit Rate (CBR). The Cell Transfer Delay (CTD) and Cell Delay Variation (CDV) of CBR traffic is of course very low (typically some tens or a few hundreds of microseconds) and peak rate policing is an adequate mechanism for controlling this traffic. Variable Bit Rate (VBR) traffic may exhibit greater CTD and CDV (in the order of milliseconds) and will in addition require the use of a sustainable cell rate policing algorithm and a maximum burst tolerance. Two types of VBR traffics are defined; real-time and non

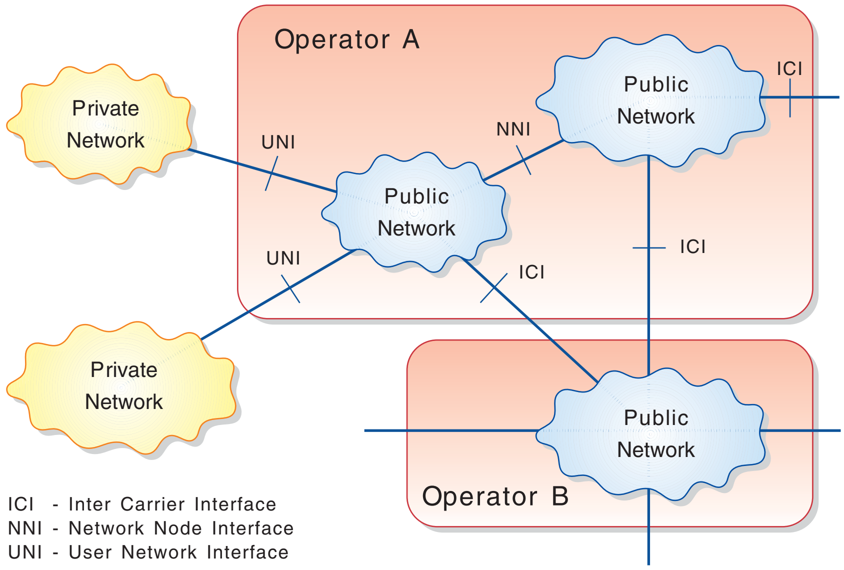


Figure 1 ATM broadband network and relevant performance measurement points

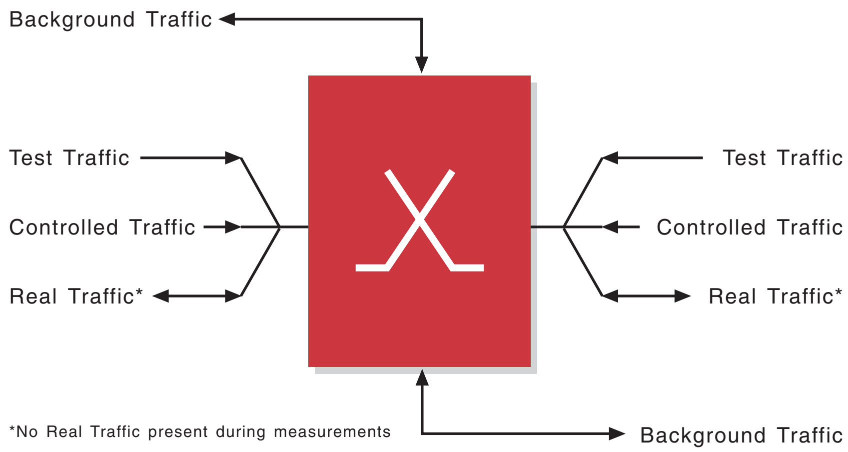


Figure 2 ATM Forum model for systems under test

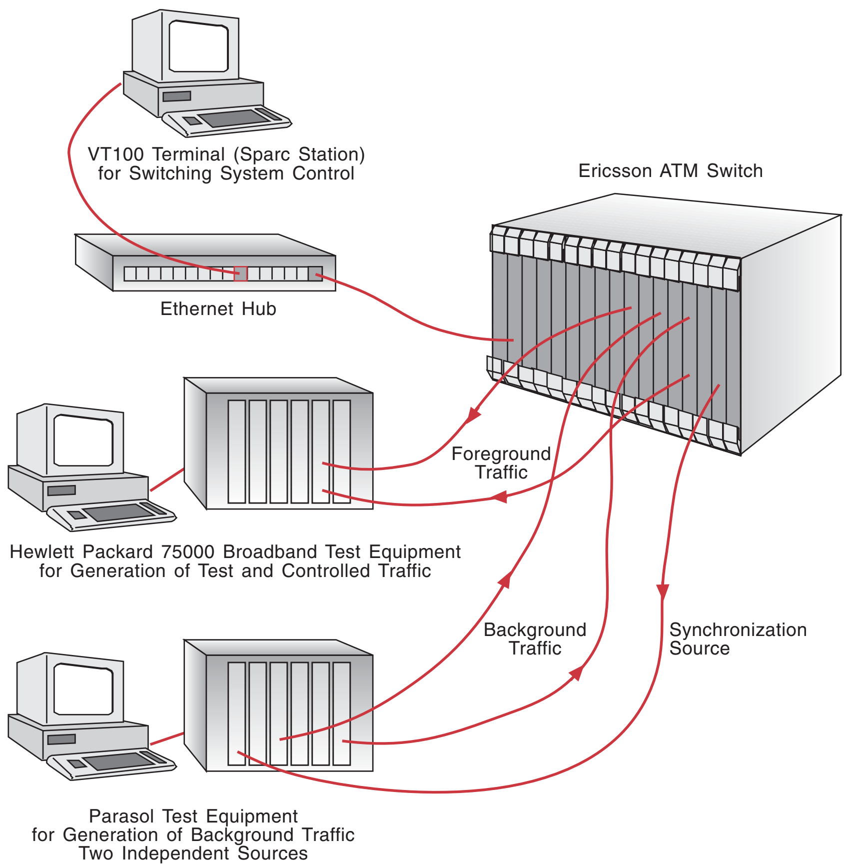


Figure 3 Test set-up

real-time VBR with the latter exhibiting less stringent requirements on CTD and CDV. Available Bit Rate (ABR) traffic may be seen as being the most extreme where CTD and CDV are concerned. For ABR traffic a closed-loop feedback control system is employed so that traffic may be sent only when there is bandwidth available. ABR traffic makes no delay guarantees.

ATM Forum is presently considering the VBR+ service which like ABR employs a closed-loop feedback control system, in addition to offering delay and loss guarantees. Another traffic type which is in the process of being drafted by ETSI is the ATM Block Transfer (ABT). Two types of ABT traffic are being suggested; immediate transfer (ABT/IT), and delayed transfer (ABT/DT).

1.4 Test traffic for an ATM network

ATM Forum defines a Quality of Service (QoS) measurement arrangement as shown in the figure below. (See [9].) Although ATM Forum states that only the UNI is of interest when performing these QoS measurements, the NNI and especially the ICI are here also considered to be of interest.

As the tests performed on the Ericsson ATM switch were out-of-service measurements, no real traffic was present during the test periods.

The test traffic is used for all measurements, while the controlled traffic is used either to influence the test traffic or to fill the available bandwidth found on the same interface. The sum of the traffic constitutes the foreground traffic and is of a deterministic nature.

Background traffic must be of a statistic nature and of sufficiently high bandwidth in order to influence the test traffic.

1.5 The test set-up

A view of the test set-up used for all of the performance measurements can be seen in Figure 3.

The Ericsson ATM switch was controlled using a VT100 window on a Sparc station. As a synchronisation source, the switch may either accept a 2 MHz clock, extract a clock from the incoming STM-1 signal, or run on a built in 155.52 MHz clock. In order to ensure total system synchronisation the incoming STM-1 signal was used to lock the switch synchronisation to the foreground traffic generator (Hewlett Packard), and the background traffic generator used an STM-1 signal from the switch as a synchronisation source.

Test traffic and controlled traffic were generated using the Hewlett Packard broadband test equipment which was chosen for the deterministic foreground traffic. For generating suitable background traffic the PARASOL test equipment developed during RACE project R1083 was used. This equipment can emulate statistical background traffic such as Bernoulli traffic as required.

1.6 Traffic profiles

Due to the asynchronous nature of ATM, all performance parameters in ATM networks have load dependent components. A complete mathematical analysis of load conditions in an ATM network is practically impossible at this time. There are just too many variables, such as the pseudo random arrival times of ATM cells, endless variations of network configurations, and all the different error rates which must be taken into account.

The previous chapter dealt with the various traffic types needed when testing an ATM switching system. No real traffic was admitted into the system while tests were ongoing. Instead, test traffic and controlled traffic were admitted into the network. The test traffic having passed through the network was then evaluated.

Background traffic was also allowed to pass through the network in order to achieve more realistic traffic load situations.

Of course, if we idealise our world such as we so often do, we may make certain estimations which we may assume to be realistic. These estimations enable us to find more or less appropriate models to be used.

In telephony, Poisson processes are used to model traffic. Poisson processes are relatively simple and easy to model, and the superposition of independent Poisson processes results in a new Poisson process which is the sum of the individual processes.

In the testing of broadband networks Bernoulli traffic has been defined by several network operators to simulate load conditions. The Bernoulli traffic used for testing the Ericsson system is described in the following chapter.

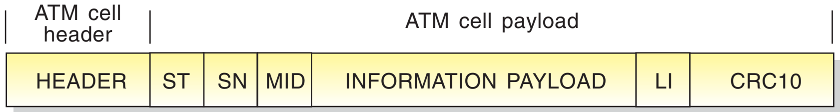


Figure 4 ATM cell supporting AAL-3/4 protocol structure

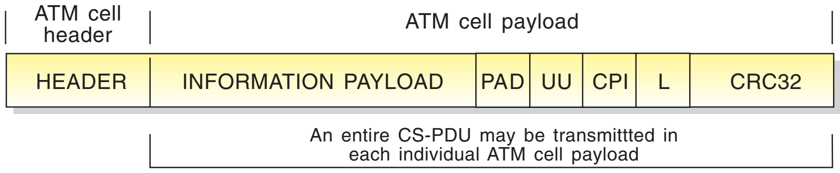


Figure 5 ATM cells supporting AAL5 protocol structure

1.7 Bernoulli traffic

The Bernoulli process, like the Poisson process, is a renewal process. Being very simple, it can be explained as the discrete-time analogue of a Poisson process. If the probability of the arrival of a cell in any time slot is denoted p, then for k time slot, the corresponding number of arrivals is binomial and may be expressed as:

$$P\{N_k = n\} = \binom{k}{n} p^n (1-p)^{k-n}$$

for values of n from 0 to k.

For more information on Poisson or Bernoulli events see "The statistical analysis of series of events" by Cox, P.A. Lewis [12].

2 Performance parameters

At the measuring points of interest depicted in Figure 1 various cell outcomes may occur. The cell may be successfully transferred, it may arrive correctly, but have an errored information field, it may not arrive successfully (become lost), or a misinserted cell may arrive at a measuring point which is not identified as belonging to any of the established connections.

Further, a sequence of errored cells, these being either errored, lost or misinserted, may be observed in a received cell block. This is termed a "severely errored cell block outcome" and the reason for this measurement is to exclude such occurrences from measurements used to evaluate the ATM layer performance. When the physical layer is functioning correctly, severely errored cell block occurrences should be extremely rare.

From these cell outcome events a number of cell performance parameters have been defined; cell error ratio, cell loss ratio, cell misinsertion rate, cell transfer delay and cell delay variation. These parameters are explained in turn in the following chapters.

The performance parameters explained in the following chapters are as defined in the standards and recommendations referred to (5, 9 and 10), but the methods for obtaining them are those found by the author to be most suitable when taking the various measurement factors into consideration.

2.1 Cell error ratio

The Cell Error Ratio (CER) is defined as the ratio between the number of erroneous cell information fields to the total number of cells transmitted in a defined period of time. Various CER measurement techniques have been looked at.

Traditional methods for calculating bit error ratios have been to insert a pseudo random bit sequence (PRBS) into the information field of the carrying frame or protocol. Cell losses which may of course occur during the measurement period will lead to a multiplication of bit errors. Instead, ATM adaptation layer (AAL) protocols were used to determine the number of bit errors in the ATM cell information field. Two different AAL protocols have been looked at.

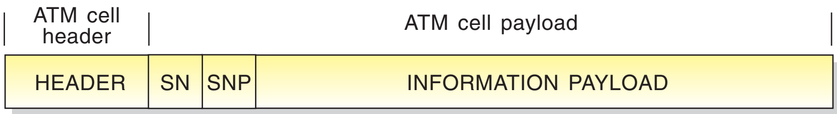


Figure 6 ATM cells supporting the AAL1 protocol

2.1.1 Measuring the cell error ratio using the AAL-3/4 protocol

In order to calculate the CER a test traffic of ATM cells with AAL3/4 protocol should be generated. An overview of this protocol structure is shown in Figure 5. See [6] for more information on the AAL3/4 protocol.

The AAL-3/4 protocol includes a cyclic redundancy code (CRC)-10 field which should detect single bit errors in the entire ATM cell payload. The error detection is performed on a cell basis, therefore lost or misinserted cells should not influence the number of CRC errors (i.e. the number of errored cells) detected.

During this measurement period the following primary measurement values were recorded:

- Selected cell bandwidth (used to derive the total number of user cells)
- Duration of measurement period
- Number of CRC-10 errors.

The following secondary measurement values were also recorded for verification of the functionality:

- Number of corrected headers (single bit header errors)
- Number of bad headers (multiple bit header errors of multiple consecutive single bit header errors)
- Number of lost cells.

By calculating the relationship Number of CRC10 Errors to Total Number of User Cells we obtain a figure for the cell error ratio. If a secondary measurement value was found to be larger than expected, then the measurement was deemed invalid.

2.1.2 Measuring the cell error ratio using the AAL5 protocol

In order to calculate the cell error ratio a test traffic of ATM cells with AAL5 protocol should be generated. An overview of this protocol structure is shown in Figure 5.

The AAL5 protocol includes a CRC-32 field which should detect single bit errors in the entire ATM cell payload. The error detection is performed on a protocol data unit (PDU) basis, but by knowing the PDU length (i.e. the number of ATM cells used to transport each PDU) we can here too calculate the cell error ratio. In addition, if we generate PDUs which are carried by a single cell (i.e. as shown in the figure above) then the likelihood of multiple errors occurring in a single PDU is reduced.

During this measurement period the following primary measurement values must be known or recorded:

- Selected Cell bandwidth (used to derive the total number of cells)
- Duration of measurement period
- Number of CRC-32 errors
- PDU length (number of cells used to carry each PDU).

The following secondary measurements values were also recorded for verification of the functionality:

- Number of corrected headers (single bit header errors)
- Number of bad headers (multiple bit header errors of multiple consecutive single bit header errors)
- Number of lost cells.

By calculating the relationship Number of CRC32 Errors to Total Number of Cells and having knowledge about the length of the segmentation and reassembly (SAR)-PDU we obtain a figure for the cell error ratio. If a secondary measurement value was found to be larger than expected, then the measurement was deemed invalid.

2.2 Cell loss ratio

The cell loss ratio (CLR) is defined as the ratio between the number of cells lost to the number of user cells transmitted in a defined period of time.

At least three possible causes of cell loss are identified. The cell delineation mechanism, which is based on the verification of the header error control (HEC) field may correct single bit errors and detect multiple bit errors. Double bit errors lead to a cell discarding, while more severe errors may lead to even worse conditions such as cell misinsertion. Due to the statistical nature of ATM traffic, buffer overflow may also occur. Finally, the user parameter control (UPC) function may discard cells if the agreed traffic contract parameters are violated.

A fourth condition which will cause cell loss is the occurrence of severely errored cell blocks, though this condition should be filtered from any measurement period. Due to the difficulty in filtering this occurrence and the fact that these should occur relatively seldom, measurements may be necessary to perform several times if the results indicate that such an event has taken place.

Normally, severely errored cell blocks should cause an Out of Cell Delineation (OCD) event and result in a Loss Of Cell Delineation (LOC) alarm. By measuring the total duration in the OCD state during the measurement period (if such an event should occur), we may calculate the number of cells which should have passed during the OCD state and exclude these from the measurement results.

Two methods of measuring cell loss are possible; cell losses detected using the sequence number, and cell losses detected using cell blocks. Because this second method was not supported by the test equipment obtained, the first method was chosen. Two different protocols for testing the CLR have been looked at and are described in the following sections.

2.2.1 Measuring the cell loss ratio using the AAL-1 protocol

In order to calculate the cell loss rate a test traffic of ATM cells with AAL-1 protocol should be generated. An overview of this protocol structure is shown in Figure 6.

The AAL-1 protocol includes a Sequence Number (SN) field and a Sequence Number Protection (SNP) field, thereby providing a very reliable cell surveillance mechanism. Lost cells should only remain undetected when a multiple of eight cells disappear. Though if a multiple of eight cells are lost the switching system should also report a Loss Of Cell Delineation and this may constitute a severely errored cell block (i.e. should not be included in the measurement).

During this measurement period the following primary values should be recorded:

- Selected Cell bandwidth
- Duration of Measurement Period
- Number of Sequence Number Errors.

Also the following secondary measurement values should be recorded:

- Number of Corrected Headers
- Number of Bad Headers.

The ratio of Lost Cells to Total Number of User Cells transmitted then gives us the Cell Loss Ratio.

2.2.2 Measuring the cell loss ratio using the AAL-3/4 protocol

The AAL-3/4 protocol may also be used for this measurement. The AAL3/4 protocol includes a four bit sequence numbering thereby reducing the chances of having a multiple number of cell losses occurring undetected (i.e. eight consecutive user cells), but does not include a protection field as in the AAL-1. Therefore use of the AAL-1 protocol is seen to be more reliable.

A schematic view of the AAL-3/4 protocol structure may be seen in Figure 5.

During this measurement period the same values as those used when using the AAL1 protocol described in chapter 2.2.1 should be recorded.

2.3 Cell misinsertion rate

Cell Misinsertion Rate (CMR) is defined as the total number of misinserted cells observed during a specified time interval divided by the time interval duration. Here too, severely errored cell blocks and their duration are to be excluded from the measurement.

Normally, cell misinsertion is assumed to be related to the occurrence of multiple bit errors (more than two) in the ATM cell header. Single bit errors are usually corrected and double bit errors are detected, but detection cannot be guaranteed with three or more errors in the cell header. For the cell to be defined as misinserted, the resulting erroneous header value must be the same as the value corresponding to another connection. Otherwise, depending upon where the error should occur, the cell will not be permitted through the switching system. A high CMR is considered as being most critical as it will also influence the QoS of other users, possibly many other users. The CMR should therefore be very low. A value of 10^{-13} has been suggested by various service providers.

No support for recording events of cell misinsertion were found in the test equipment. Therefore the CMR was recorded by logging the number of unidentifiable cells registered by the switching system.

2.4 Cell transfer delay

Cell Transfer Delay (CTD) is the time taken for a cell to be transferred from source to destination. The CDT may vary greatly depending on the switch load. An increasing CTD may be seen as an early indication of possible cell loss due to buffer overflow conditions in the network.

In order to test the average cell delay through the switching system, calibration of the measurement equipment must first be performed. The HP broadband test equipment is calibrated by first measuring the cell delay introduced when the coaxial cable from the test equipment (output port) is looped back to the input port.

The average cell delay registered during this measurement must then be subtracted from the values obtained during measurements through the switching system. Because the CTD may vary during load conditions, this measurement should be made a number of times with varying background traffic.

Alternatively, the time stamping function of the test equipment may be used. The HP test equipment used, supported the time stamping of cells.

2.5 Cell delay variation

This chapter discusses issues to do with the measuring of Cell Delay Variation (CDV) and the introduction of CDV for test purposes. Two different parameters for CDV measurements are defined by ITU-T; 1-point CDV which is the difference between the actual time of arrival of a cell and the expected cell arrival time, and 2-point CDV which measures the actual delay variation of a cell stream. 2-point CDV measurements are performed using special test cells and can only be performed out-of-service.

In order to test that the UPC algorithm does not discard cells which conform to the traffic contract and has a cell delay variation below some predetermined value, at first glance a very difficult measurement is needed, like the one shown in Figure 7 in which we need to tune background traffic such that a sufficient amount of CDV is introduced. Further-

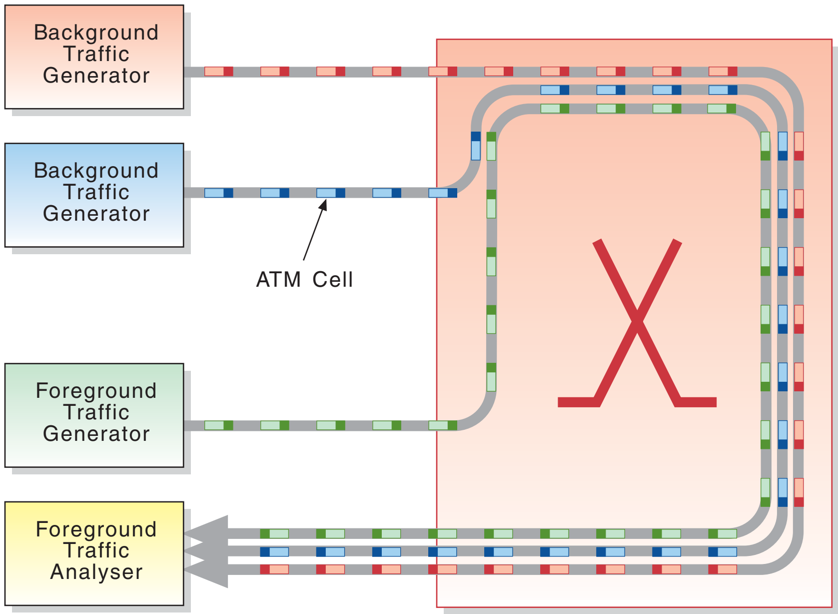


Figure 7 Illustrating some of the complexities with introducing CDV for test purposes

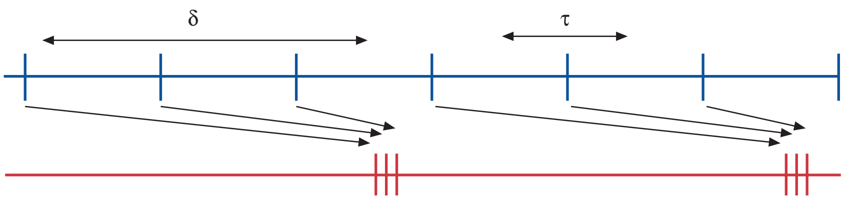


Figure 8 Deterministic introduction of CDV

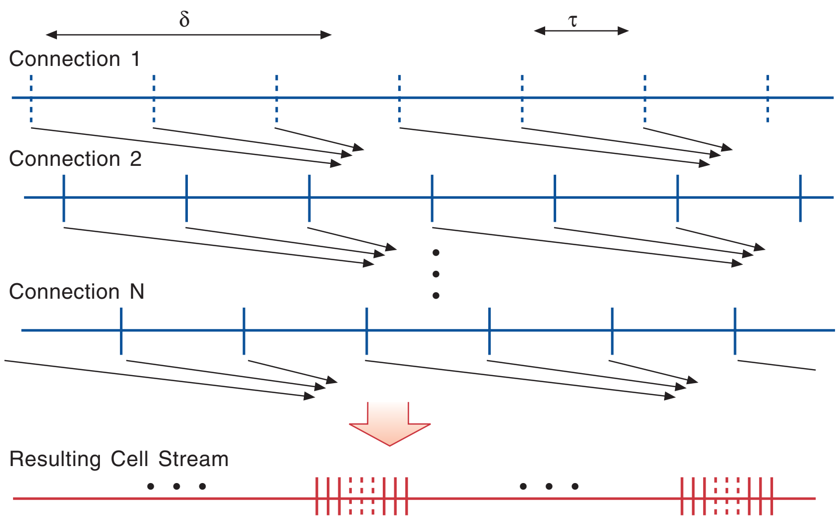


Figure 9 Test traffic with deterministic CDV introduced

more, prior to entrance in the analyser we need the UPC algorithm to test that conforming cells are not lost.

This is a very complicated task, and instead we suggest the following much simpler alternative outlined in the following sections.

It should be noted that the following description applies to the Ericsson policing algorithm which in the strict sense is not a peak cell rate (PCR) scheme, alhough the algorithm implemented does in fact satisfy all of the criteria set by the network operators which are using the system. If a strict PCR mechanism is to be tested then the clusters of cells should be reduced to two.

2.5.1 Testing of CDV for a single connection

Let the test traffic be deterministic traffic constructed in the following manner:

Consider initially a CBR cell stream with interarrival time T between cells (upper line in 8). Since the CDV tolerance is d, take now the first cell and delay it by the allowed amount d. Since cell sequence integrity must be maintained also the next $\left\lfloor \frac{\delta}{T} \right\rfloor$ cells must be delayed, and delay them just so much that $\left\lfloor \frac{\delta}{T} \right\rfloor + 1$ cells are coming back to back. Let cell no. $\left\lfloor \frac{\delta}{T} \right\rfloor + 2$ have no delay but at cell no. $\left\lfloor \frac{\delta}{T} \right\rfloor + 3$ you start to repeat the whole procedure.

Thereby a periodic sequence (the lower one at the figure) is generated which can be programmed into the HP traffic generator.

In the UPC algorithm of the switching system the leak rate should be set such that it exactly decrements a single cell arrival to zero in T time units. The threshold value should be put to

$2 + \left\lfloor \frac{\delta}{T} \right\rfloor$ in accordance with the argument given in [2].

2.5.2 Testing of CDV at the STM-1 rate

To generate a CDV of 300 µs (which was the network operator requirement) it has been explained that bursts of 3 and 3 cells for a single connection must be sent from the traffic generator.

In order to reduce the time necessary to perform these tests involving CDV limitations, either a single connection of sufficiently high bandwidth must be involved in the measurement, or several connections which together have the desired bandwidth must be measured upon.

Obviously, there is an upper limit to the bandwidth of a single connection which may be measured with the appropriate CDV introduced. Therefore the latter of these two measurements is seen to be simpler to achieve using the available traffic generator and should reduce the test period.

A graphical representation of this idea is presented in Figure 9.

The above traffic can be generated using a memory based traffic generator such as the Hewlett Packard Broadband Test Equipment.

3 Test results

Before the performance measurements outlined in this report were made, requirements for the various values were given by some network operators. At least one network operator presented the following requirements:

- The CLR should be less than 10^{-10} measured over a 12 hour period.
- The CER should be less than 5 x 10^{-10} measured over a 12 hour period.
- The CMR should be less than 10^{-13}. (For practical reasons it was stated that only one misinserted cell was allowed to be recorded during a measurement period of 12 hours.)
- The average CTD was to be less than 250 usec.
- The CDV was to be less than 300 usec with a probability of $1 - 10^{-10}$.

3.1 Measured results

Initial tests showed unacceptable values for many of the above measurements. After having made some simple measurements with an Ohmmeter, insufficient grounding of the equipment was found to be the cause. This greatly increased the performance, but in order to achieve the results given above the synchronisation set-up as explained in section 1.5, "The test set-up" was necessary.

Due to company policy detailed information about the measured values is not given in this document. What can be said is that all of the performance requirements detailed in the previous chapter were met.

For many of the measurement intervals where CLR, CER and CMR were measured, not a single erroneous event was recorded.

4 Validity of tests and test results

All of the tests outlined in the paper were based on requirements by network operators. The interfaces used, bandwidth allocated during test periods, duration of test period, and the desired results were specified a priori by network operators. How to obtain the values for each measurement were specified by the test group where the tests were performed.

Much can be said about the validity of each test, and there is little doubt that improvements to the tests outlined are possible. Test equipment is constantly becoming more advanced, as is the equipment it is being designed to test. These tests were made with the equipment available at the time of testing.

The duration of the measurements is also a subject for discussion. When attempting to record rare events such as the CLR, CER and CMR, months or even years may be necessary. All we can do in the time frame available is make a estimate which will be correct with a certain probability. This issue and some other areas which may be subject to improvement are outlined in the following section.

4.1 Validity of the test set-up

As previously stated, due to several reasons the tests outlined in this report were to be kept simple. Firstly, limitations in both the test equipment and the system under test (for simplicity only three switch interfaces were used) presented a finite number of test configuration possibilities.

It was stated that 80 % Bernoulli background traffic was to be present in the switch during the measurement periods. Bernoulli traffic should have been present on all switch interfaces simultaneously, though whether this would have had any impact on the measurement results is doubtful.

Secondly, with the short time available for performing each test and the need to perform each test a number of times, simplicity was essential.

4.2 Validity of long measurements

A very important question which has arisen during the initial stages of these tests is; "What measurement duration is necessary in order to be able to estimate the frequency of rare events, such as the CLR, with reasonable accuracy". Though requirements on the duration of each test were given by network operators, the following section outlines some of the thoughts around this issue.

4.2.1 Rare events and point processes

Let us assume that during some (long) experiment a number of (rare) events are observed and counted. The process of events is described by the mathematical term *point process*, and a point process can be described in two ways; either by the *interval* representation or by the *counting* representation. In the interval representation, track is kept of all interarrival times, and this gives in some sense a microscopic and very detailed view of the process. In the counting representation, a measurement interval is chosen and the number of events in this interval is studied. This yields in some sense a more macroscopic view of the process.

A basic objective of the experiment is to estimate the rate (or the probability) of rare events. The chances of doing this is very dependent on the a priori knowledge we have of the process.

If the process is known, more can be said. If the process under study is a Poisson process with rate λ, and we take a measurement interval of length T then the mean number of events will be λT and the standard deviation will be $\sqrt{\lambda T}$. This implies that with a measurement period of 100 times the expected time between events we will be able to say that the number of events will be between 80 and 120 with approximately 95 % probability. If the measurement period is only 4 times the expected time between events there will be no events at all with 2 % probability and with 95 % probability there will be between 1 and 8 events, i.e. we have no accurate estimation of the rate.

If the underlying process is renewal (the same as Poisson except that the interarrival time between events is arbitrary and not exponential), then the more variance the interarrival time distribution has, the less can be said. For example, if the variance is 4 times the case for the Poisson process the measurement period of 100 expected interarrival times will give a 95 % confidence interval of 60 to 140 events.

As this small example shows, the variance of the underlying process plays a significant role on the time a measurement must run before a given accuracy of the expected rate of rare events can be given.

If the process is unknown things are even worse, because it is very unlikely that the process of the rare events are known in advance. The cell loss process in ATM switch buffers is not well understood and depends very much on the traffic which is even less understood.

One way to attack the problem is to run a number M (at least 10) experiments and try to arrange the experiments such that they are independent of each other. If we let X_i denote the number of events in experiment i, then we have the following estimates of the mean and variance

$$\overline{X} = \frac{1}{M}\sum_{i=1}^{M} X_i$$

$$\overline{S} = \frac{1}{M-1}\sum_{i=1}^{M}(X_i - \overline{X})^2$$

Since the X_i's are independent we may approximate their sum by a normal distribution and obtain quantiles from that.

This method of course has the problem that the variance is also only an estimate, and for processes with correlation in time the uncertainty in the variance estimate may be significant. The interested reader is recommended to see [12].

4.2.2 Rare events in the test measurements

As explained previously, the bandwidth of the test traffic during the CLR, CER and CMR measurements was set to 80 % of the synchronous digital hierarchy (SDH) payload capacity. This results in approximately 282,566 cells/second. It is then simple to calculate that during a 12 hour period approximately 1,220,685 x 10^{10} cells will pass through the switch during this time. With a CLR of 10^{-10} this means roughly that only one cell is allowed to be lost during each test period.

Because of this, each test had to be run a number of times. At least 10 times is suggested here. This will significantly increase the reliability of the performance measurements made.

5 Future testing

The tests performed on the Ericsson system were mainly performed in order to verify the performance of equipment being delivered for integration into the pan-European pilot project. As this was an ATM network in its first stage of deployment, the requirements and tests were maintained at a simplest possible level. Only a peak rate bandwidth allocation scheme was implemented and semi-permanent connections were established. These semi-permanent connections were used to offer a Virtual Leased Line (VLL) service.

During the second stage, equipment will need to offer a much wider range of services. Switched connections must be supported and several different bandwidth allocation schemes will be required. This will necessitate complete compliance with the ITU-T recommendations I.356, I.371 and I.610 (among others). For example, compliance with these recommendations will in the future enable us to perform in-service performance monitoring.

In-service measurements should give us much better information about the network performance during *real* load conditions. These measurements may also be made without the need for external test equipment. Measurements may be made at any time, during any traffic loads and over any duration of time.

Although these recommendations are not yet finalised, much is done. Ongoing and future tests will contribute information required to complete these recommendations within the standardisation organisations.

6 Concluding remarks

This paper has attempted to show that simple measurements can be made in order to verify the performance of an ATM switch or network. Before the testing was started certain requirements regarding the CLR, CER, CMR, CTD, and CDV were obtained from network operators wishing to purchase the Ericsson equipment.

Various methods of measuring the cell performance are explained with both the test set-up, the protocols used and the parameters necessary to record are given.

The test results are presented and then their validity has been discussed. When discussing the validity, focus is placed upon the fact that rare events are difficult to measure accurately. Due to this difficulty in measuring rare events, it was stated that these tests should be performed a number of times.

In conclusion, the test results indicated that the switch under test did conform to the requirements specified.

7 References

1 Onvural, R O. *Asynchronous transfer mode networks : performance issues.* Artech House Inc., 1994. ISBN 0-89006-622-0.

2 COST. *Information technologies and sciences: performance evaluation and design of multiservice networks, final report.* 1991. (COST 224.) ISBN 92-826-3728-X.

3 ITU. *Error performance parameters and objectives for international constant bit rate digital paths at or above the primary rate.* Geneva 1994. (ITU-T Rec. G.826.)

4 ITU. *General aspects of quality of service and network performance in digital networks, including ISDNs.* Geneva, 1993. (ITU-T Rec. I.350.)

5 ITU. *B-ISDN ATM layer cell transfer performance.* Geneva 1993. (ITU-T Rec. I.356.)

6 ITU. *B-ISDN ATM adaptation layer (AAL) specification.* Geneva 1993. (ITU-T Rec. I.363.)

7 ITU. *Traffic control and congestion control in B-ISDN.* Geneva 1994. (ITU-T Rec. I.371.)

8 ITU. *B-ISDN operation and maintenance principles and functions.* Geneva 1994. (ITU-T Rec. I.610.)

9 *Introduction to ATM Forum test specifications (Draft 1.1).* ATM Forum, October 4, 1994.

10 *B-ISDN ATM layer cell transfer : performance parameters.* ANSI T1.511-1994, March 24, 1994.

11 Frost, V S, Melamed, B. Traffic modelling for telecommunications networks. *IEEE communications magazine,* March 1994.

12 Cox, Lewis, P A. *The statistical analysis of series of events.* Methuen, 1966.

The effect of end system hardware and software on TCP/IP throughput performance over a local ATM network

BY KJERSTI MOLDEKLEV, ESPEN KLOVNING AND ØIVIND KURE

Abstract

High-speed networks reinstate the end-system as the communication path bottleneck. The Internet TCP/IP protocol suite is the first higher-level protocol stack to be used on ATM based networks. In this paper we present how the host architecture and host network interface are crucial for memory-to-memory TCP throughput. In addition, configurable parameters like the TCP maximum window size and the user data size in the write and read system calls influence the segment flow and throughput performance. We present measurements done between Sparc2 and Sparc10 based machines for both generations of ATM-adapters from FORE Systems. The first generation adapters are based on programmed I/O; the second generation adapters on DMA. To explain the variations in the throughput characteristics, we put small optimized probes in the network driver to log the segment flow on the TCP connections.

1 Introduction

The TCP/IP (Transmission Control Protocol/Internet Protocol) stack has shown a great durability. It has been adapted and widely used over a large variety of network technologies, ranging from low-speed point-to-point lines to high-speed networks like FDDI (Fiber Distributed Data Interface) and ATM (Asynchronous Transfer Mode). The latter is based on transmission of small fixed-size cells and aims at offering statistical multiplexing of connections with different traffic characteristics and quality of service requirements. For ATM to work as intended, the network depends on a characterization of the data flow on the connection. TCP/IP has no notion of traffic characteristics and quality-of-service requirements, and considers the ATM network as high-bandwidth point-to-point links between routers and/or end systems. Nevertheless, TCP/IP is the first protocol to run on top of ATM.

Several extensions are suggested to make TCP perform better over networks with a high bandwidth-delay product [1]. At present, these extensions are not widely used. Furthermore, in the measurements to be presented in this paper, the propagation delay is minimal making the extensions above of little importance.

The TCP/IP protocol stack, and in particular its implementations for BSD UNIX derived operating systems, has continuously been a topic for analyses [2], [3], [4], [5], [6], [7]. These analyses consider networks with lower bandwidth or smaller frame transmission units than the cell-based ATM network can offer through the ATM adaptation layers, AALs.

This paper contributes to the TCP analyses along two axes. The first is the actual throughput results of TCP/IP over a high-speed local area ATM network for popular host network interfaces and host architectures. Measurements are done on both Sparc2 and Sparc10 based machines using both generations of ATM network interfaces from FORE Systems; the programmed I/O based SBA-100 adapters with segmentation and reassembly in network driver software, and the more advanced DMA based SBA-200 adapters with on-board segmentation and reassembly. Both the hardware and software components of the network interface, the network adapter and the network driver, respectively, are upgraded between our different measurements.

The second axis is an analysis of *how* and *why* the hardware and software components influence the TCP/IP segment flow and thereby the measured performance. The software parameters with the largest influence are the maximum window size and the user data size. In general, the throughput increases with increasing window and user data sizes up to certain limits. It is not a monotonous behavior; the throughput graphs have their peaks and drops.

TCP is byte-stream oriented [8]. The segmentation of the byte stream depends on the user data size, the window flow control, the acknowledgment scheme, an algorithm (Nagle's) to avoid the transmission of many small segments, and the operating system integration of the TCP implementation. The functionality and speed of the host and network interface also influence the performance; more powerful machines and more advanced interfaces can affect the timing relationships between data segments and window updates and acknowledgments.

The rest of this paper is outlined as follows: The next section describes the measurement environment and methods. The third section presents in more detail the software and hardware factors influencing the performance; the protocol mechanisms, the system environment factors, and the host architecture. The fourth section contains throughput measurements and segment flow analysis of our reference architecture, a Sparc2 based machine using the programmed I/O based SBA-100 interface. The fifth section presents performance results when upgrading a software component, namely the network driver of the network interface. The sixth section discusses the results when upgrading the hosts to Sparc10 based machines. The throughput results and segment flows using the SBA-200 adapters in Sparc10 machines follow in the seventh section. The paper closes with summary and conclusions.

2 Measurement environment and methods

The performance measurements in this paper are based on the standard TCP/IP protocol stack in SunOS 4.1.x. We used two Sparc2 based Sun IPX machines and two Sparc10 based Axil 311/5.1 machines. The I/O bus of both machine architectures is the Sbus [9] to which the network adapter is attached. The Sun machines run SunOS 4.1.1, while the Axil machines run SunOS 4.1.3. For our TCP measurements the differences between the two SunOS versions are negligible. Access to the TCP protocol is through the BSD-based socket interface [11]. The workstations have both ATM and ethernet network connections. Figure 1 illustrates the measurement environment and set-up.

2.1 The local ATM network

The workstations are connected to an ATM switch, ASX-100, from FORE Systems. The ASX-100 is a 2.5 Gbit/s bus-based ATM switch with an internal Sparc2 based switch controller. The ATM physical interface is a 140 Mbit/s TAXI interface [12]. The ATM host network interfaces, SBA-100 and SBA-200, are the first and second generation from FORE.

The first generation Sbus ATM adapter, SBA-100 [17], [18], is a simple slave-only interface based on programmed I/O. The ATM interface has a 16 kbyte receive FIFO and a 2 kbyte transmit FIFO. The SBA-100 network adapter performs on-board computation of the cell based AAL3/4 CRC, but the segmentation and reassembly between frames and cells are done entirely in software by the network driver. The SBA-100 adapters have no hardware support for AAL5 frame based CRC. Therefore, using the AAL3/4 adaptation layer gives the best performance. The SBA-100 adapters were configured to issue an

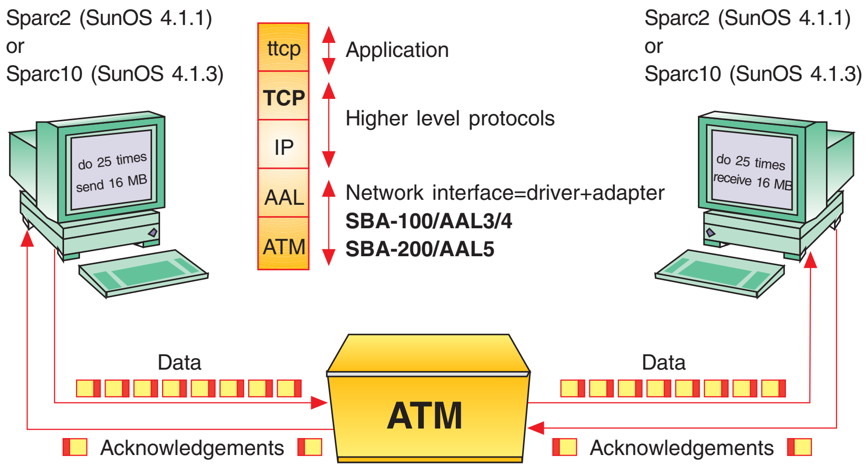


Figure 1 Measurement environment and set-up

AAL3/4 end-of-message interrupt on incoming packets.

The second generation Sbus adapter, SBA-200 [18][19], includes an embedded Intel i960 RISC control processor, and hardware support for both AAL 3/4 and AAL5 CRC calculation. It is an Sbus master device and uses DMA for data transfer on both the send and receive path. The segmentation and reassembly processing is performed by the control processor on the adapter using the host memory for storage of frames. The SBA-200 adapters have the best performance when using AAL5, and they were configured to provide an AAL5 end-of-frame interrupt on incoming packets. The ATM transmission capacity is not a bottleneck. Therefore, compared to AAL3/4 the reduced AAL5 segmentation and reassembly header overhead should not be a decisive factor on the measured throughput. Using AAL3/4, the on-adapter receive control processor turns out to be a bottleneck for large TCP window sizes.

2.2 Host architectures

The memory-to-memory measurements presented in this paper are performed between two Sparc2 based Sun IPX machines, and between two Sparc10 based Axil 311/5.1 machines. The workstations were not tuned in any way, except for allowing only a single user during the measurements. All standard daemons and other network interfaces were running as normal. In the rest of this paper we name the two machine types Sparc2 and Sparc10.

The MIPS rating (28.5 vs. 135.5) is about 4.5 times higher for the Sparc10 compared to the Sparc2. Apart from the CPU, the main difference between these machine architectures is the internal bus structure. While the CPU in the Sparc2 has direct access to the Sbus, the Sparc10 has separated the memory (Mbus) and I/O (Sbus) bus. Thus, access to the Sbus from the CPU must pass through an Mbus-to-Sbus interface (MSI) which certainly increases the latency when accessing the network adapter.

2.3 Measurement methods

We used the *ttcp* application program to measure the TCP memory-to-memory throughput. The ttcp program uses the write and read system calls to send and receive data. We modified the ttcp program to set the socket options which influence the maximum TCP window size on a connection. Each measured throughput point is the average of 25 runs. Each run transfers 16 Mbyte memory-to-memory between the sender and the receiver.

The CPU utilization is measured for different window sizes with the user data size set to 8192 bytes. The SunOS maintenance program *vmstat,* was run to register the average CPU load. For minimal interference with the throughput measurements, vmstat was run every 10 seconds in parallel with a transfer of 256 Mbyte between the sender and receiver.

To analyze the segment flow on the ATM connections, we put small optimized probes in the network driver to log all packets on the ATM connections of dedicated runs. One log corresponds to one run which transfers 16 Mbytes. The probes parse the TCP/IP packets and register events in a large log table during the data transfer. Included in each event log is a time stamp, an event code and a length field. The time stamp is generated using the SunOS uniqtime() kernel function which accesses the internal µsec hardware clock. The length field is used to log, among other things, the announced window size, the TCP packet length, and sequence numbers as indicated by the event code. The contents of the log table is printed off-line using the *kvm* library functions available in SunOS 4.1.x.

The probes were put in the network driver to only log traffic on the ATM network. To minimize the logging overhead the probes were placed on the machine with the least CPU utilization. Thus, for the SBA-100 adapters logging was done on the send side while logging was done on the receive side for the SBA-200 adapters. The throughput results with and without the logging mechanism indicate the logging overhead to be less than 5%.

The log information is presented in primarily four kinds of graphs. The first three, covering the whole measurement period, present the size of transmitted segments, the number of outstanding bytes, and the receiver announced window size. Due to the granularity of the x-axis, fluctuations along the y-axis show up as black areas. The fourth kind of graph uses a finer time resolution and covers only a small interval of the connection life-time. Two values are displayed in the same figure; the announced window size and the number of outstanding unacknowledged bytes. The transmission of a segment is shown as a vertical increase in the number of outstanding bytes, while an acknowledgment is displayed as a drop.

3 Factors influencing performance

There are several factors which influence the segment flow on a TCP connection. In addition to the protocol itself, the application interface and the operating system [10], the CPU and machine architecture, and the underlying network technology affect the segment flow. In this section we describe and quantify some of these factors.

3.1 Environment and implementation factors

The interface to the TCP/IP protocol in BSD-based systems is through the socket layer [11]. Each socket has a send and a receive buffer for outgoing and incoming data, respectively. The size of the buffers can be set by the socket options SO_SNDBUF and SO_RCVBUF. In SunOS 4.1.x the maximum size of these buffers is 52428 bytes. The *user data size* is the size of the message or user buffer in the write/read system calls. The *segment size* is the user data portion of a TCP/IP packet.

On transmit, the user data size is the number of bytes which the write system call hands over to the socket layer. The socket layer in SunOS 4.1.x copies maximum 4096 bytes of the user data into the socket send buffer before TCP is called. If the user data size is larger than 4096 bytes, TCP is called more than once within the write system call. When there is no room in the socket send buffer, the application accessing the protocols through the socket interface does a sleep to await more space in the socket send buffer.

On receive, the user data size is the byte size of the user buffer in which data is to be received. The read system call copies bytes from the socket receive buffer and returns either when the user buffer in the system call is filled, or when there are no more bytes in the socket receive buffer. However, before the system call returns, the socket layer calls TCP, which checks if a window update and acknowledgment should be returned.

Figure 2 presents how the segment flow may depend on the send user data size. The segment size is the payload size of the TCP/IP packet. The figure is based on logs from Sparc10 machines with SBA-100/2.2.6 interfaces. For a window size of 8 kbytes, Figure 2 (a) shows a snap-shot of the segment flow on a connection with a user data size of 8192 bytes. Figure 2 (b) shows the same, but with a user data size of 8704 bytes. In Figure 2 (a), the write system call is called with a user data size of 8192 bytes. First, 4096 bytes are copied into the socket layer. Then, TCP is called and a 4096 byte segment is transmitted on the network. This is repeated with the next 4096 bytes of the 8192 byte user data size. In Figure 2 (b) there are 512 bytes left over after two segments of 4096 bytes have been transmitted. These bytes are transmitted as a separate 512 byte TCP segment. The last segment flow is clearly less efficient, because it does not fill the window in-between each acknowledgment, and it has a higher per-byte overhead.

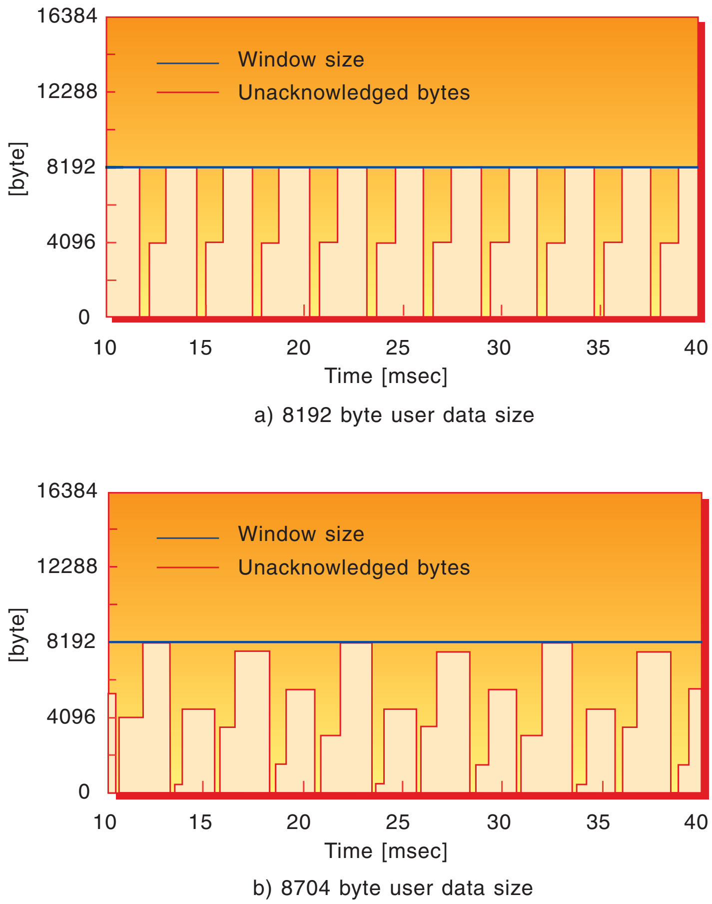


Figure 2 Segment flow depends on user data size

3.2 TCP protocol factors

TCP is a byte stream oriented protocol, and does not preserve boundaries between user data units. TCP uses a sliding window flow control mechanism. Hence, the window size is relative to the acknowledgment sequence number. In SunOS 4.1.x a window update is sent if the highest announced window sequence number edge will slide at least twice the maximum segment size, MSS, or if the highest advertised window sequence number edge will slide at least 35% of the maximum socket receive buffer [11], [13]. The *available* space in the socket receive buffer reflects the TCP window which is announced to the peer. The size of the socket receive buffer is the maximum window which can be announced.

The TCP maximum segment size, MSS, depends on the maximum transmission unit, MTU, of the underlying network [14]. For our ATM network the MTU is 9188 bytes and TCP computes the MSS to 9148 bytes (the MTU size less the size of the TCP/IP header). The slow-start algorithm [15] is not an issue in our measurements since the sender and receiver reside on the same IP subnetwork. Nagle's algorithm [16] was introduced as a solution to the "small-packet problem" which results in a high segment overhead if data is transmitted in many small segments. Sending a new TCP segment which is smaller than MSS bytes and smaller than half the announced window size is inhibited if any previously transmitted bytes are not acknowledged.

The TCP delayed acknowledgment strategy piggybacks acknowledgments on either data segments or window updates. In addition, acknowledgments are generated periodically every 200 ms. An incoming acknowledgment releases space in the socket send buffer. The reception of an acknowledgment may therefore trigger transmissions of new segments based on the current number of bytes in the socket send buffer.

The timer generated acknowledgment occurs asynchronously with other connection activities. Therefore, such an acknowledgment may not adhere to the window update rules above. As a conse-

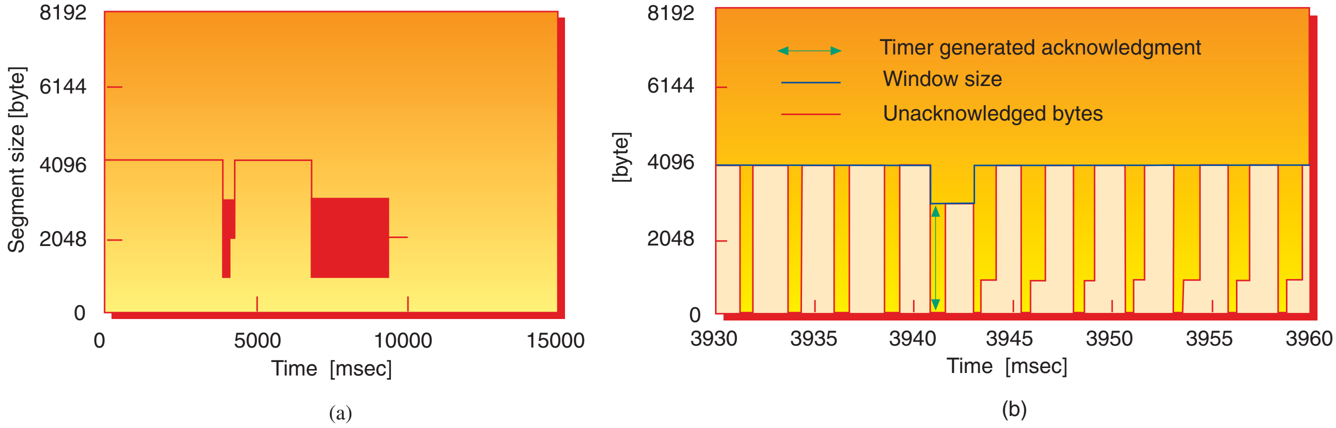


Figure 3 Timer generated acknowledgments

quence, a timer generated acknowledgment can change the segment flow on the connection. An example of this effect is found in Figure 3 (a) which displays a log (Sparc2, SBA-100/2.2.6) of the size of the segments on a connection with a 4 kbyte window and a user data size of 4096 bytes. There are three different flow behaviors on the connection; 4096 byte segments, fluctuation between 1016 and 3080 byte segments, and fluctuation between 2028 and 2068 byte segments. Initially, 4096 byte segments are transmitted. After nearly 4 seconds a timer generated acknowledgment acknowledges all outstanding bytes and announces a window size of 3080 bytes. This acknowledgment is generated before all bytes are copied out of the socket receive buffer, and the window is thereby reduced corresponding to the number of bytes still in the socket receive buffer. This affects the size of the following segments which will fluctuate between 1016 and 3080 bytes. Figure 3 (b) presents the segment flow before and after this timer generated acknowledgment which acknowledges 4096 bytes and announces a window of 3080 bytes. A similar chain of events gets the connection into a fluctuation of transmitting 2028 and 2026 byte segments. This shows up as the horizontal line of the last part of the segment size graph in Figure 3 (a).

3.3 Host architecture and network adapters

To establish how the difference in the Sparc2 and the Sparc10 bus architecture influences the achievable performance we measured the time for the send and receive paths for both architectures. Using the SBA-100 adapters, the measured driver times are proportional to the segment size. The receive times of the SBA-100 adapter include the time it takes to read the cells from the receive FIFO on the adapter. The corresponding send times include the time it takes to write the cells to the transmit FIFO on the adapter. Using the SBA-200 adapters the measurable driver times are more or less byte independent. Obviously, the driver times for the DMA-based SBA-200 adapter do not include the time to transfer the segment between host memory and the network adapter memory. (We do not have an Sbus analyzer.) Figure 4 presents for different segment sizes for both Sparc2 and Sparc10 the total send and receive times and the driver send and receive times as seen from the host:

- the *total send time* is the time from the write call is issued to the driver is finished processing the outgoing segment,
- the *driver send time* is the time from the driver processing starts until it is finished processing the outgoing segment.
- the *total receive time* is the time from the host starts processing the network hardware interrupt to the return of the read system call, and
- the *driver receive time* is the time from the host starts processing the network hardware interrupt until the packet has been inserted into the IP input queue.

Each measurement point is the average of 1000 samples. A client-server program was written to control the segment flow through the sending and receiving end system. The client issues a request which is answered by a response from the server. Both the request segment and the response segment are of the same size. The reported send and receive times are taken as the average of the measured send and receive times at both the client and the server. To be able to send single segments of sizes up to MSS bytes, we removed the 4096-byte copy limit of the socket layer (Section 3.1).

As expected, the receive operation is the most time-consuming. The total send and receive processing times of the Sparc10 are shorter for all segment sizes compared to the Sparc2. However, using the SBA-100 adapters, the driver processing times are in general faster on the Sparc2. (The only exception is for small segments.) This is due to the fact that the Sparc10 CPU does not have direct access to the Sbus. The latency to access on-adapter memory is thereby longer. Thus, a 4.5 times as powerful CPU does not guarantee higher performance with programmed I/O adapters. As mentioned above, the SBA-200 driver processing times do not include the moving of data between the host memory and the network adapter. The Sparc10 SBA-200 driver send times are longer than the driver receive times. For Sparc2 it is the other way round. On transmit the driver must dynamically set up a vector of DMA address-length pairs. On receive, only a length field and a pointer need to be updated. The Sparc2 must in addition do an invalidation of cache lines mapping the pages of the receive buffers, while the Sparc10 runs a single cache invalidation routine. The Sparc10 SBA-200 driver send time is slightly longer than the corresponding Sparc2 times, as the Sparc10 sets up DVMA mappings for the buffers to be transmitted.

The send and receive processing times reflect the average time to process one single segment. The processing times of the receive path do not include the time from the network interface poses an interrupt until the interrupt is served by the network driver. Neither do the times include the acknowledgment generation and reception. The numbers are therefore

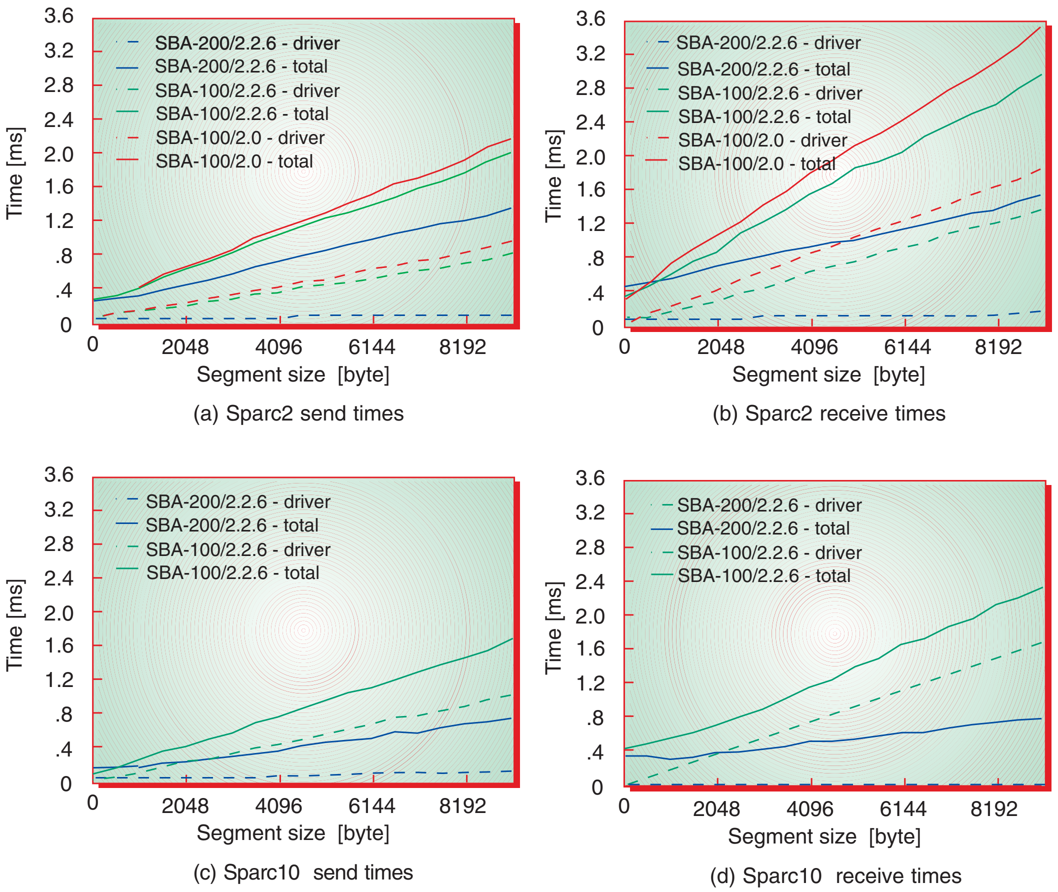


Figure 4 Segment send and receive processing times

not fully representative for performance when the protocol operates like stop-and-go.

In order to explain the Sparc2 and Sparc10 relative performance for smaller window sizes when using SBA-100 adapters, we measured the throughput with TCP operating as a stop-and-go protocol for these configurations. This was achieved by setting the window size equal to the user data size. The throughput under such a behavior is shown in Figure 5 for both host architectures. Up to a user data size between 2048 and 2560 bytes the slower Sparc2 has a higher throughput. This is most likely due to the longer Sbus access time combined with the relatively longer interrupt time.

4 Initial throughput measurements

This section presents the initial throughput measurements for the reference architecture, the Sparc2 with the SBA-100 network adapter with the first network driver version, i.e. version 2.0. Figure 6 (a) presents the measured TCP end-to-end memory-to-memory throughput for different user data sizes and different window sizes. The throughput dependent on window size for a fixed user data size of 8192 bytes is presented in Figure 6 (b), and the corresponding CPU utilization in Figure 6 (c). Both the user data size and the window size affect measured throughput. From the graphs it is clear that increasing the TCP window size above 32 kbytes has no increasing effect on the throughput. On the contrary, such an increase in window size results in a significantly varying throughput dependent on user data size. The reason is cell loss at the receiving host which causes TCP packet loss and retransmissions of lost packets.

In general, the receive operation is known to be more time consuming than the transmit operation. At the receive side, there are among other things demultiplexing and scheduling points and interrupts which do not occur at the send side. In addition, reading from memory is more time consuming than writing to memory. Clearly, this is evident from the driver processing times presented in Figure 4. Furthermore, the processor (CPU) utilization histograms in Figure 6 (c) show the receiver to be heavier loaded than the sender. When the TCP connection starts losing packets due to cell loss at the receiver, both the throughput and CPU utilization are degraded. TCP employs positive acknowledgment with go-back-n retransmission. The sender fills the window, and if every byte is not acknowledged, it relies on retransmission timers to go off before it starts sending at the point of the

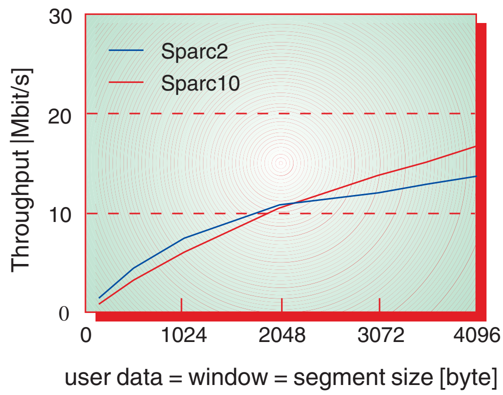


Figure 5 Sparc2 versus Sparc10 segment processing, SBA-100/2.2.6

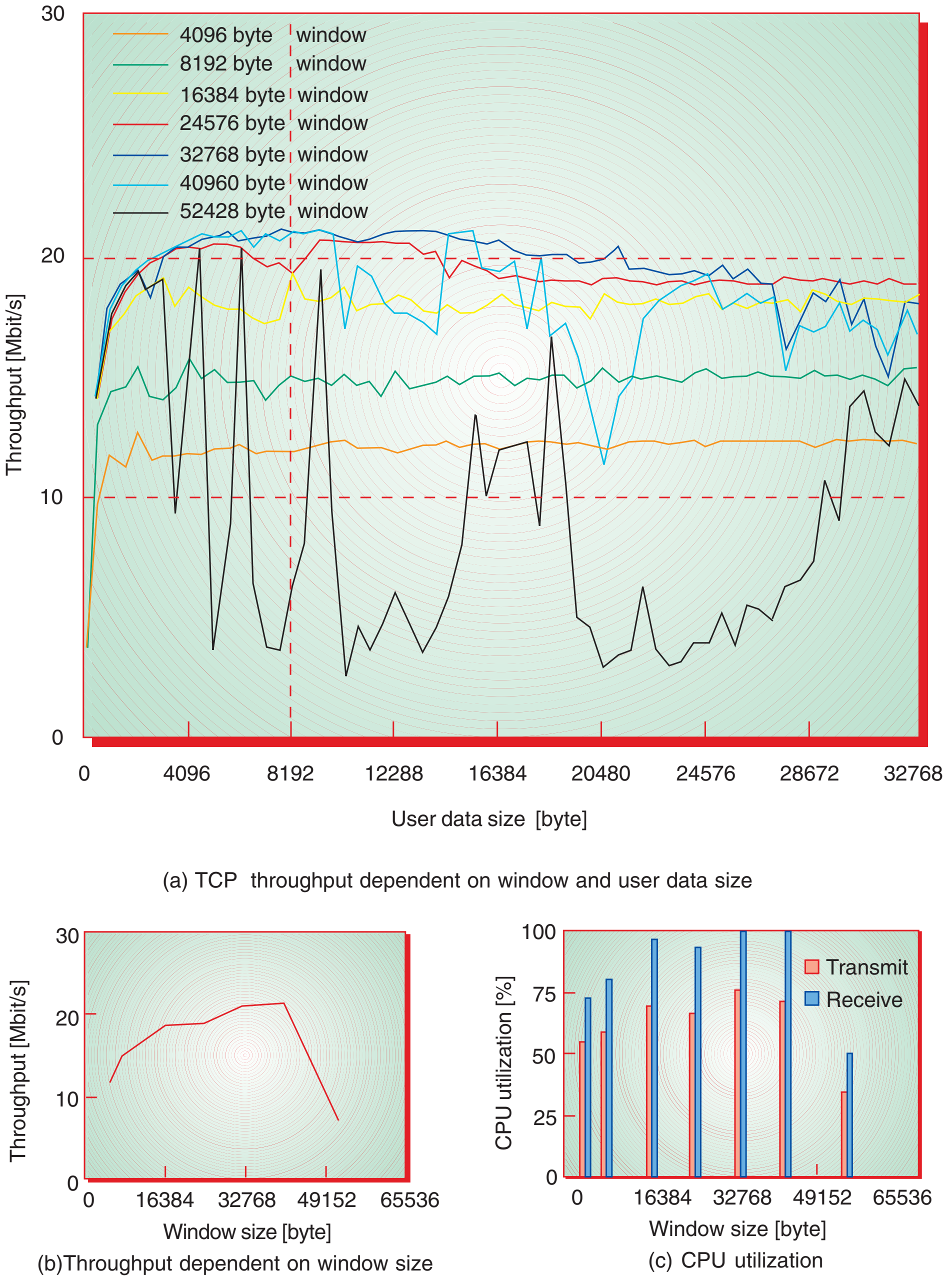


Figure 6 Sparc2 SBA-100, network driver version 2.0

first unacknowledged byte. Because TCP relies on the retransmission timers to detect packet losses, the end systems will have idle periods in-between the packet processing.

The general characteristics of the graphs in Figure 6 can be summarized as follows

- The maximum performance is about 21 Mbit/s. For an 8192-byte user data size the measured throughputs dependent on window size are: 12 Mbit/s, 15 Mbit/s, 19 Mbit/s, 19 Mbit/s, 21 Mbit/s, 21 Mbit/s, and 6 Mbit/s.
- Up to a window size of 32–40 kbytes (dependent on the user data size), the larger the window size, the higher the measured throughput. This is as expected, as an increase in window size will utilize more of the processing capacity of the host.
- Window sizes above 32–40 kbytes (dependent on the user data size) cause cell loss and thereby packet loss at the receiver. This is evident through the degradation in measured throughput.
- The receiver is heavier loaded than the sender.

5 Throughput measurements with upgraded network driver

In this section we discuss the throughput graphs for the Sparc2 with the SBA-100 network interface, and the FORE network driver version 2.2.6. Thus, a software upgrade is the only change compared to the measurements in the previous section. The main difference between the two network driver versions is the access to the transmit and receive FIFO on the network adapter. The 2.2.6 version transports data more effectively to and from the network adapter through using double load and store instructions to access the FIFOs. Thus, bus burst transfers of 8 instead of 4 bytes are used to transfer cells to and from the network interface.

For different window sizes Figure 7 (a) presents the measured throughput performance dependent on user data size. The throughput dependent on window size for an 8192 byte user data size, and the corresponding CPU utilization are shown in Figure 7 (b) and Figure 7 (c), respectively.

The general characteristics of the graphs in Figure 7 can be summarized as follows:

- The maximum performance is approximately 26 Mbit/s. For an 8192-byte user data size the measured throughputs dependent on window size are approximately: 14 Mbit/s, 18 Mbit/s, 22 Mbit/s, 23 Mbit/s, 26 Mbit/s, 25 Mbit/s, and 25 Mbit/s.
- Due to a more effective communication between the host and the network adapter, there is no cell loss at the receive side which causes dramatic throughput degradations. Compared to the 2.0 driver version, the ratios of the driver segment receive and send times are smaller for the 2.2.6 version. This can be computed from the times presented in Figure 4.
- A window size above 32 kbytes does not contribute much to increase the performance. It is now the end system processing and not the window size which is the bottleneck.
- Generally, the larger the window size, the higher the measured throughput. However, using an arbitrary user data size, an increase in window size may not give a performance gain. Depend-

ent on the window size the user data size giving peak performance varies.

- The form of the throughput graph shows that the measured throughput increases with increasing user data sizes. Thus, the segment dependent overhead is reduced when transmitting fewer segments. After an initial phase, the measured throughput is more or less independent of increasing the user data size. However, for larger window sizes, the measured throughput decreases with larger user data sizes.
- All window sizes have their anomalies, i.e. peaks and drops, for certain user data sizes.
- The receiver is heavier loaded than the sender.

5.1 Throughput dependent on user data size

From Figure 7 (a) it is evident that the throughput is not indifferent to the user data size. For a given number of bytes to be transferred, the smaller the user data size of the write system call, the more system calls need to be performed. This affects the achievable throughput in two ways. One is due to the processing resources to do many system calls, the other is due to a lower average size of the protocol segments transmitted on the connection. Therefore, for small user data sizes increasing the user data size will increase the throughput. The increase in throughput flattens when an increase in user data size does not significantly influence the average segment size of the connection.

For 4k, 8k, and 16k window sizes the throughput is more or less independent of large user data sizes. It is the window size and host processing capacity which are the decisive factors on the overall throughput. On the contrary, larger window sizes experience a light degradation in throughput. The throughput degradation is caused by acknowledgments being returned relatively late. The larger the receive user buffer, the more bytes from the socket buffer can be copied before an acknowledgment is returned. Therefore, the average time to acknowledge each byte is a little longer, and the throughput degrades.

For small and large user data sizes the throughput may be increased if, independent of the send user data size, the size of the *receive* user buffer is fixed. For small user data sizes a larger receive buffer reduces the number of read system

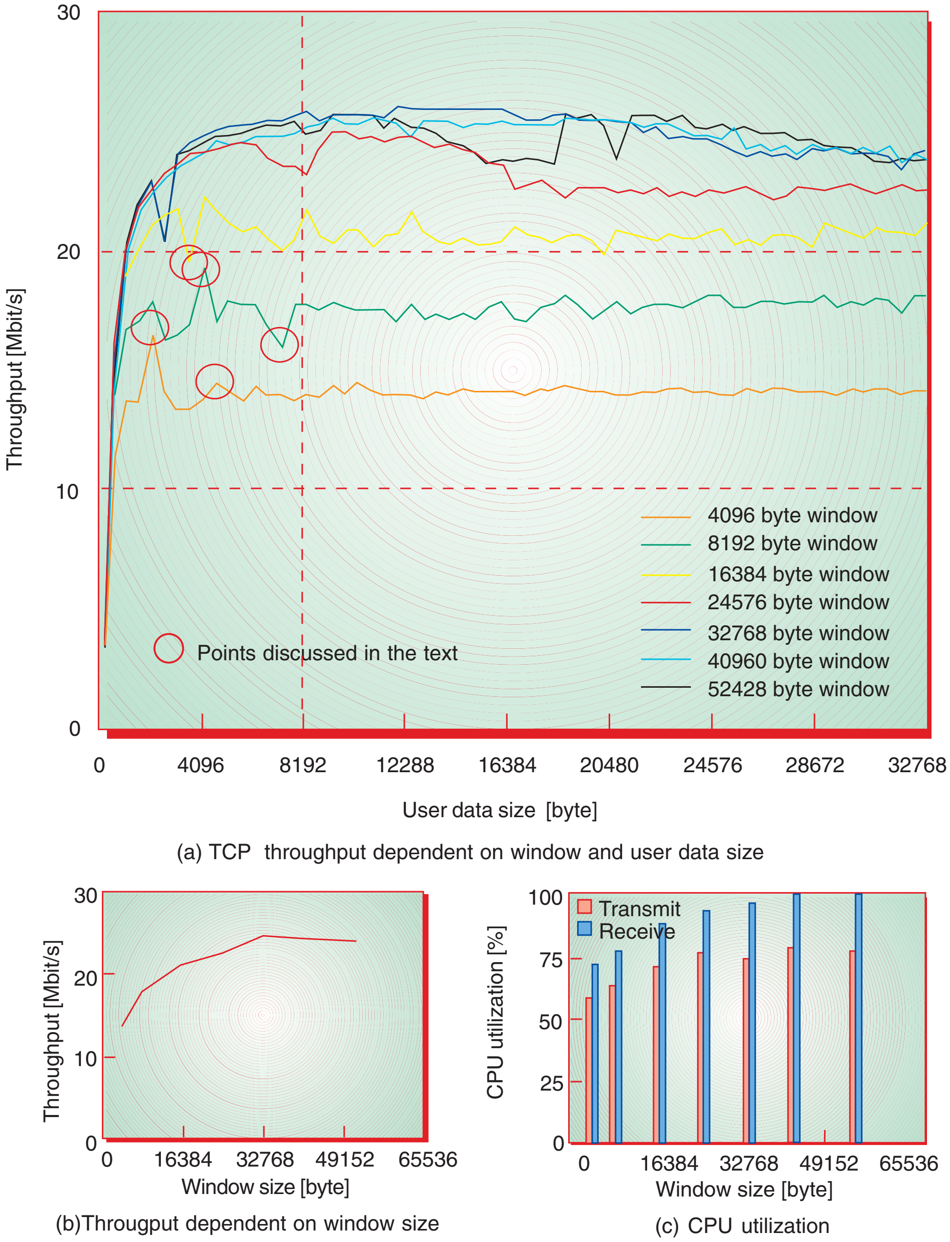


Figure 7 Sparc2 SBA-100, network driver version 2.2.6

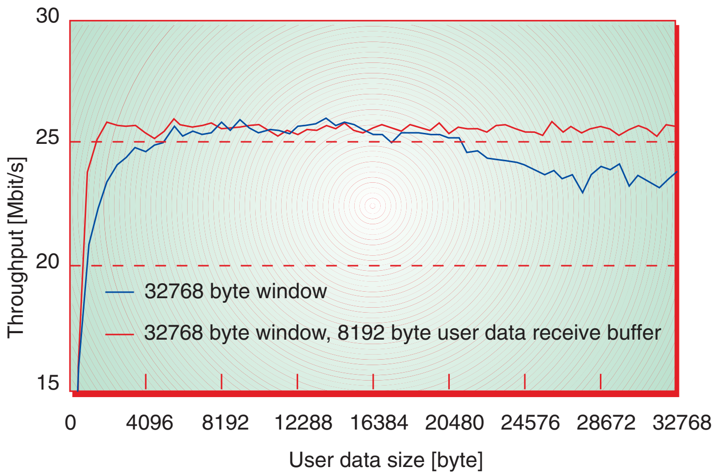


Figure 8 Throughput changes with a fixed size receive user buffer

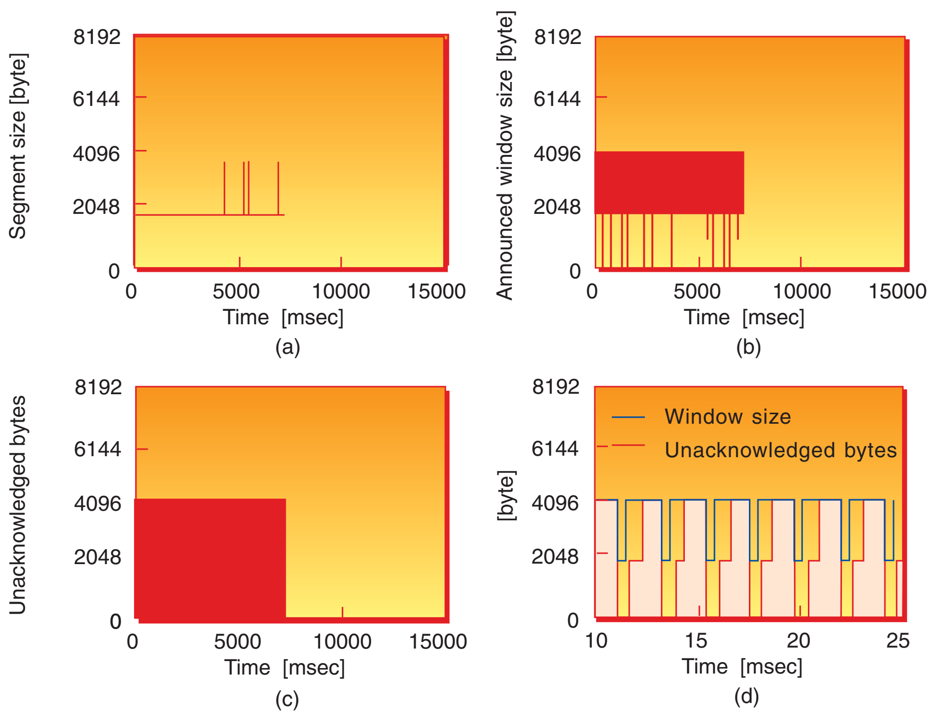


Figure 9 Segment flow with a 4 kbyte window and a 2048 byte user data size

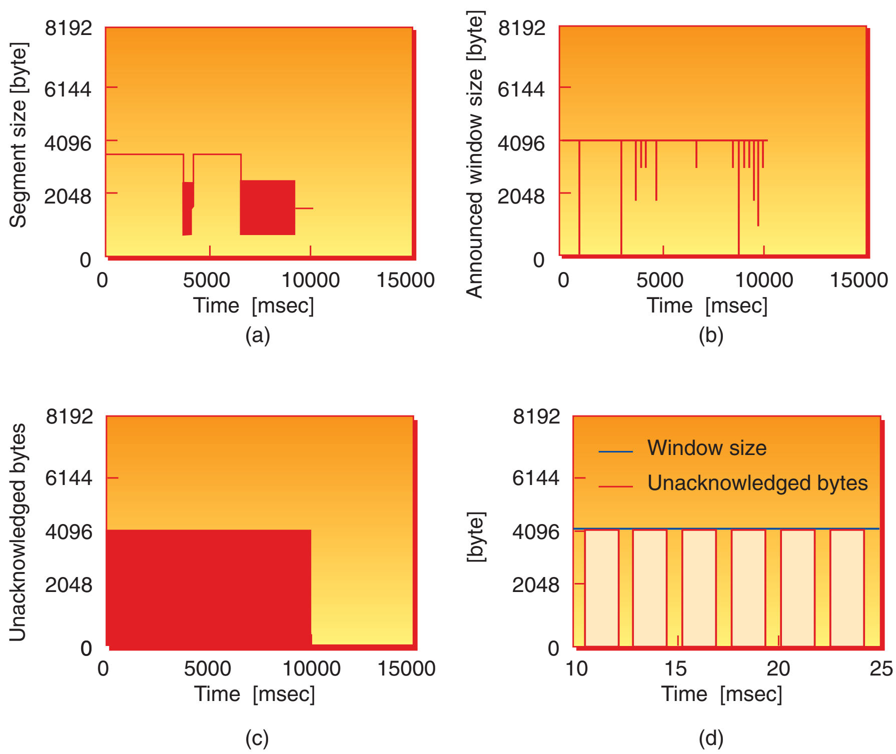


Figure 10 Segment flow with a 4 kbyte window and a 4096 byte user data size

calls and thereby processing overhead. For larger data units a smaller receive buffer makes the socket layer more often call TCP to check if a window update and acknowledgment should be returned. Thereby, the average time to acknowledge each byte is reduced and the sender byte transmission rate is increased. The measurements in Figure 7 (a) are done with symmetrical send and receive user buffer sizes. The throughput graph for the 32k window size is repeated in Figure 8. The throughput fixing the receive user buffer to 8192 bytes is also presented in Figure 8. As expected, the throughput is higher for both small and large user data sizes.

5.2 Throughput peaks

Figure 7 reveals parameter settings which can increase the performance by approximately 20%, e.g. with a 4 kbyte window and a 2048 byte user data size, that is, the point w4k-u2048. Other points experience a drop in performance. In short, the peaks and drops are due to the user data size strongly affecting the segment flows.

With a 4 kbyte window size, a user data size of 2048 bytes gives *the* maximum throughput. This is due to parallelism between the sender and receiver processing of segments. Another such point is w8k-u4096.

Figure 9 and Figure 10 present for a connection life time of w4k-u2048 and w4k-u4096 (a) the size of the transmitted segments, (b) the receiver announced window size, (c) the number of unacknowledged bytes, and (d) a time slice of both the window size and the unacknowledged bytes. Studying the segment flows reveals that a 2048 byte user data size causes window updates and acknowledgments to be returned more often. This reduces the average number of outstanding unacknowledged bytes, and in short, the data bytes are faster acknowledged. (Due to the maximum size of the log table in the driver Figure 9 misses the last 15% of the segments.)

Comparison of Figure 9 (a) and Figure 10 (a) reveals a clear difference in segment sizes. It is the segment flow which directly affects the measured throughput, as this decides the work load distribution on the sender and receiver. The segment flow on w4k-u2048 primarily consists of 2048 byte segments. The segment flow on w4k-u4096 starts out with 4096 byte segments. As presented in Figure 3 it is a timer generated acknowledgment that changes the segment flow.

For w4k-u2048 the announced window size fluctuates between 2 and 4 kbytes, while the announced window size of w4k-u4096 primarily is 4 kbytes, Figure 9 (b) and Figure 10 (b). Lower window size announcements are due to timer generated acknowledgments. The number of outstanding unacknowledged bytes vary on both connections between 0 and 4096 bytes, Figure 9 (c) and Figure 10 (c).

Figure 9 (d) and Figure 10 (d) present the number of unacknowledged bytes relative to the window size for a 15 ms interval of the connection life time. From these figures it is clear that a window update and acknowledgment on the average is returned faster for a 2048 relative to a 4096 byte user data size. Independent of the connection life-time, in Figure 9 (d) there are two acknowledgments/window updates per 4096 bytes sent. The first acknowledges all 4096 bytes. It thereby releases space in the send buffer so that the sender can continue its transmission. The processing at the receiver and sender overlap on the reception and the transmission of the 2048-byte segments. The sender starts processing the next 2048-byte segment while the receiver processes the previous 2048-byte segment. In addition to this processing parallelism between the two 2048-byte segments in-between acknowledgments, there is an overlap between copying data out of the socket receive buffer at the receiver side and copying data into the socket send buffer at the sender side for the segments on each side of the acknowledgments. In Figure 10 (d) a window update announces a 4 kbyte window and acknowledges 4096 bytes. Thus, there is no processing parallelism between the sender and receiver.

The higher the throughput, the higher the CPU utilization. The *vmstat* program showed that the CPU utilization of w4k-u2048 was approximately 80% and 100% for transmit and receive, respectively. The same numbers for the w4k-u4096 were 55% and 70%.

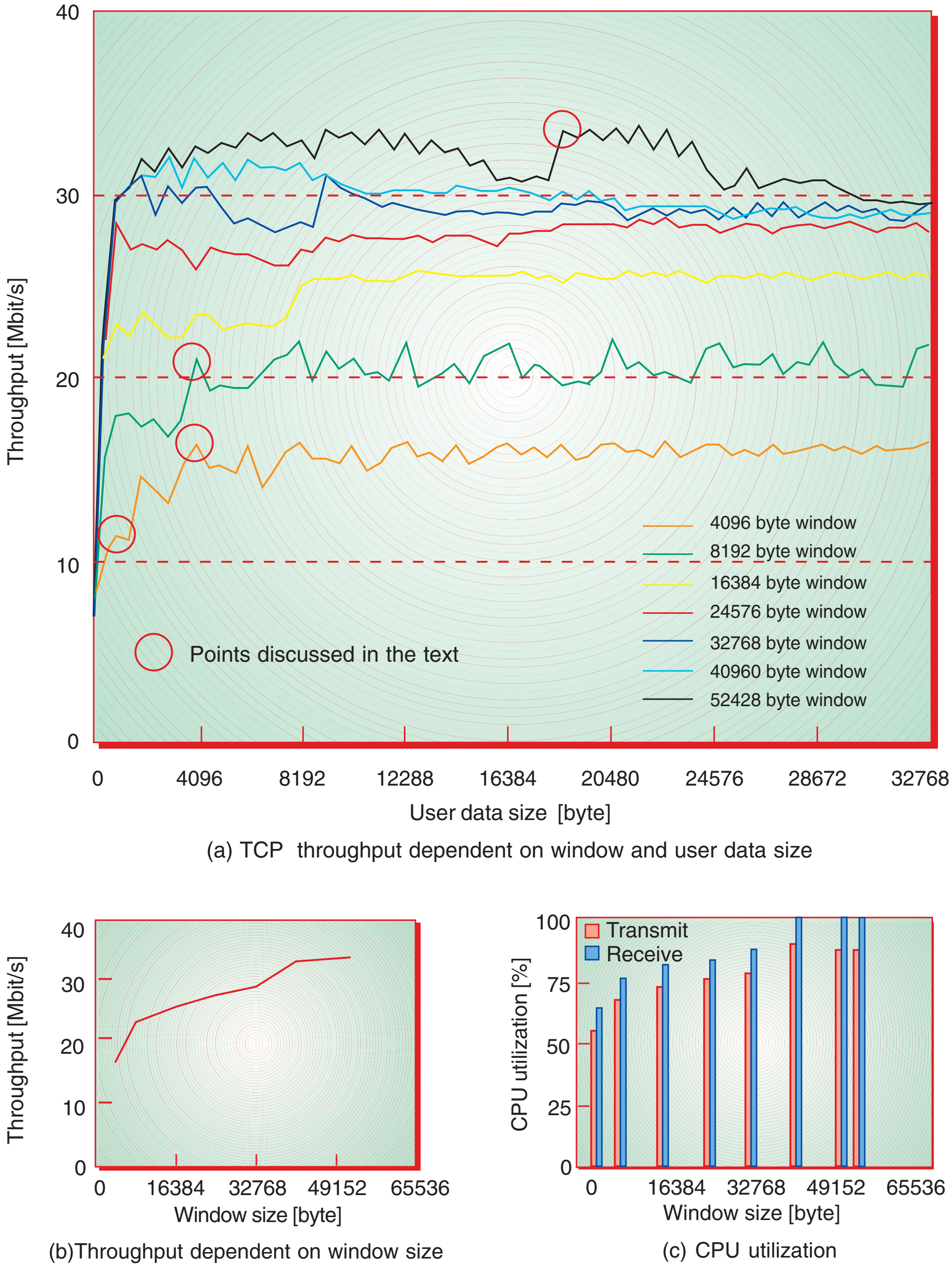


Figure 11 Sparc10 SBA-100, network driver version 2.2.6

5.3 Throughput drops

Figure 7 reveals parameter settings with a sudden drop in measured performance, e.g. w8k-u7168 and w16k-u3584. The throughput drop is due to an inefficient segment flow compared to the neighboring points. The point w8k-u7168 has a stop-and-go behavior for the whole connection life-time. In-between acknowledgments a 1024 byte and a 6144 byte segment are transmitted. Its neighboring points transmit 8192 bytes in-between acknowledgments.

Neither the points above nor the point w16k-u3584 fully utilize the announced window size. On this connection an average of about 8 kbytes are transferred in-between acknowledgments, while its neighboring points transfer on average approximately 12 kbytes.

6 Upgrading the host architecture

In this section we present measurements between the two Sparc machines using the SBA-100 adapters with the network driver version 2.2.6. For different window sizes Figure 11 (a) presents the measured throughput performance dependent on user data size. The throughput dependent on window size for an 8192 byte user

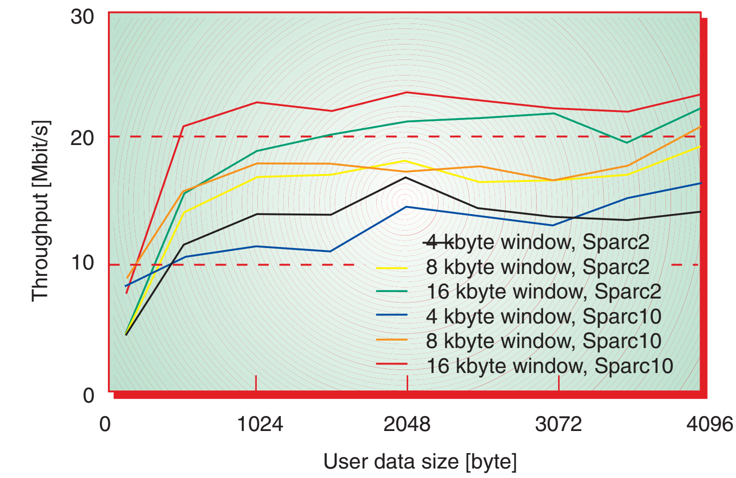


Figure 12 Sparc2 versus Sparc10 throughput, SBA-100/2.2.6

data size, and the corresponding CPU utilization are shown in Figure 11 (b) and Figure 11 (c), respectively.

The discussion of the throughput results in Figure 11 is related to the results from the previous section. The experiences on upgrading the host architecture are:

- Overall, a faster machine improves throughput performance. However, the improvement is lower than might be expected from the host MIPS numbers. The maximum throughput is approximately 34 Mbit/s. For an 8192-byte user data size the measured throughputs dependent on window size are approximately: 16 Mbit/s, 22 Mbit/s, 25 Mbit/s, 27 Mbit/s, 28 Mbit/s, 32 Mbit/s, and 33 Mbit/s.

- For small window sizes and small user data sizes the performance is actually reduced.

- Within one window size, the “flat” part of the graph is not as flat anymore.

- The receiver is heavier loaded than the transmitter. Compared to the S2 SBA-100/2.2.6 CPU utilization, the difference between the sender and receiver load is smaller. This is in correspondence with the ratio of the total segment receive and send times which is lower for the Sparc10 compared to the Sparc2.

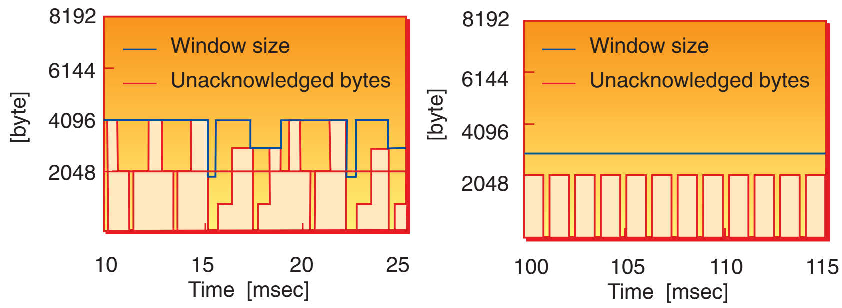
6.1 Degradation of throughput

Figure 12 repeats the throughput from Figure 7 (Sparc2) and Figure 11 (Sparc10) for small user data sizes and 4, 8, and 16 kbyte window sizes. For certain user data sizes the Sparc10 has lower performance for a 4 kbyte window. The lower throughput on the Sparc10 is due to the small segment sizes. For small segment sizes the Sparc10 is slower than Sparc2, see Figure 5.

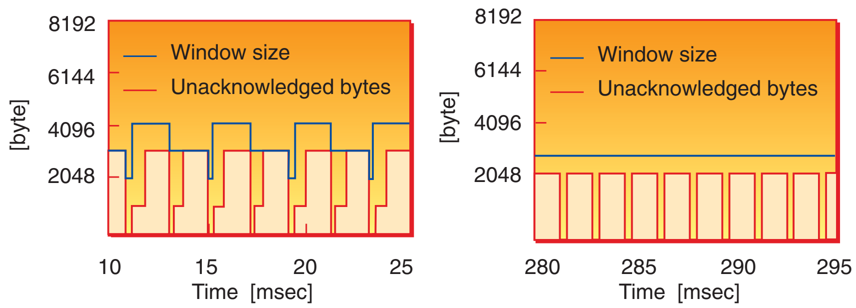
One example is the user data size equal to 2048 bytes. The segment flow (see Figure 9) on the connection is the same on both machines. Another point is a user data size of 1024 bytes and a window size of 4 kbytes. The w4k-u1024 connections have primarily two different segment flow behaviors. These are depicted for Sparc2 and Sparc10 in Figure 13 (a–b). On average, the Sparc10 has fewer outstanding bytes than the Sparc2 for both flows. On w4k-u1024 the Sparc10 more often than Sparc2 announces a reduced window size. The window reductions are due to the receiver not coping so well with the sender for small segments. Relative to their senders, the Sparc10 receiver is slower than the Sparc2 receiver. This can be deduced from the relationship between the total receive and send processing times in Figure 4 (a) and (b).

6.2 Amplification of peaks

For small window sizes, i.e. 4 and 8 kbytes, the throughput varies as much as 10% for different user data sizes. Peak throughput is reached with a user data size of an integer multiple of 4096 bytes. This creates a regular flow of 4096 byte segments on the connections. For other user data sizes, mismatch between user data size and window leads to less regular segment sizes.



(a) Sparc2 segment flows



(b) Sparc10 segment flows

Figure 13 Sparc2 versus Sparc10 segment flow at the point w4k-u1024

Corresponding peaks do not exist for all points on the Sparc2 measurements. The reason is the difference in segment flow on connections with the same parameter settings. For example, the segment flow on a w4k-u4096 connection was presented in Figure 10. Initially, w4k-u4096 connections on Sparc2 and Sparc10 have exactly the same segment flow. It was the timer generated acknowledgment which caused a change in the segment size giving a segment flow which is less efficient. This is less probable on the Sparc10, because the relative time data bytes are in the socket receive buffer, is shorter than for the Sparc2. Remember that for a 4096 byte segment the Sparc10 total receive processing time is lower, but the receive driver processing time is higher. In addition, the Sparc10 gains more relative to Sparc2 when the segment sizes are large.

6.3 Plateaus in the throughput graphs

For the largest window size the throughput starts degrading for large user data sizes. However, with a 52428 byte window at certain larger user data sizes there is a throughput plateau with a throughput increase of up to 3 Mbit/s before the throughput once more degrades. The same behavior can be observed for Sparc2 machines in Figure 7. However, the plateau is more visible for the Sparc10 because this machine utilizes the largest window size better than the Sparc2.

At w52-u18432 the throughput increases by 3 Mbit/s. At this point the user data size exceeds 35% of the maximum window size. Hence, according to the window update rules the receiver needs only one read system call before a window update is returned. Acknowledgments for outstanding bytes will arrive faster and thus increase the throughput. Figure 14 shows the effect on the segment flow by increasing the user data size from 16384 to 20480 bytes. The point w52-u16384 is before the plateau, and the point w52-u20480 is on the plateau. For w52-u20480 the receiver acknowledges fewer bytes at a time. The announced window size is also lower, because the receiver does not copy out more than 20480 bytes before an acknowledgment is returned. The improvement in throughput disappears when the user data sizes increase even further. This is due to the degradation effect explained in Section 5.1.

7 Upgrading the network adapter

In this section we present measurements using the more advanced SBA-200 network interface in the Sparc10 based machines. The network driver version is still 2.2.6. The upgrade relieves the host CPU from writing and reading single cells to/from the network interface. The interface towards the adapter is now frame based, and the adapter uses DMA to move AAL cell payloads between the host memory and the adapter itself. Since the driver processing of a TCP segment now is nearly byte-independent (see Figure 4), the number of segments is an important factor for SBA-200 adapter performance. In addition, more of the CPU processing resources can be used on the TCP/IP segment processing.

For different window sizes Figure 15 (a) presents the measured throughput performance dependent on user data size. The throughput dependent on window size for an 8192 byte user data size, and the corresponding CPU utilization are shown in Figure 15 (b) and Figure 15 (c), respectively.

The main conclusions from upgrading the network interface are:

- The overall throughput increases dramatically. The peak throughput is approximately 62 Mbit/s. For an 8192-byte user data size the measured throughputs dependent on window size are approximately: 21 Mbit/s, 30 Mbit/s, 37 Mbit/s, 41 Mbit/s, 44 Mbit/s, 55 Mbit/s, and 60 Mbit/s.
- As expected, the throughput is higher the larger the window size. However, for small window sizes, the variation in throughput between different user data sizes is high.
- For large user data sizes and large window sizes the throughput does not degrade.
- The host required *network* processing is clearly more time-consuming using the SBA-100 adapters. Using the SBA-200 adapters the hosts account primarily for the TCP/IP related overhead. Thus, the number of segments needed to transfer a certain amount of data clearly influences the achievable performance.
- The sender is heavier loaded than the receiver. This probably reflects the fact that the total send and receive time are approximately the same (see Figure 4), and it is more time consuming for the sender to process incoming acknowledgments compared to the time it takes the receiver to generate the acknowledgments.

Figure 15 (a–b) clearly illustrates the fact that increasing the window size increases the throughput. For small window sizes, more bytes transmitted in-between acknowledgments will give a performance gain. For larger window sizes, additional bytes in the socket buffer when the window update arrives, decides the throughput. For example, increasing the window from 32 to 40 kbytes gives a 10 Mbit/s throughput increase. The primary segment flow for these two window sizes is two MSSs in-between acknowledgments. With a 32 kbyte window, there is not room for another 2*MSS bytes in the socket send buffer. Thus, on reception of an acknowledgment, one MSS is directly transmitted. The next MSS cannot be transmitted before more bytes are copied into the socket buffer from the user application. However, for a 40 kbyte window there is room for more than 4*MSS bytes, thus a window update causes two

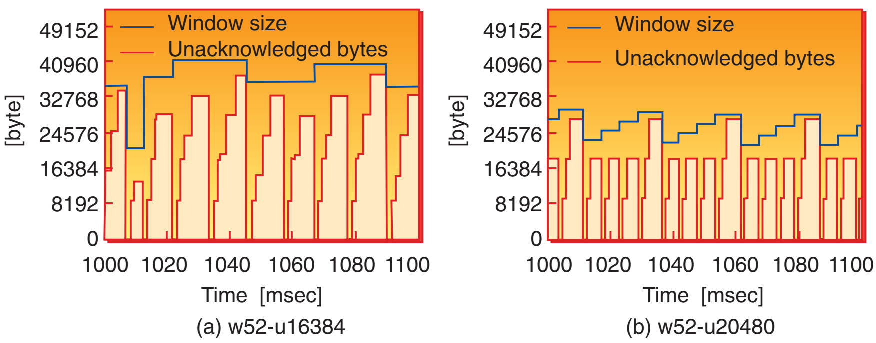


Figure 14 Sparc10 segment flow (a) before and (b) on the plateau, SBA-100/2.2.6

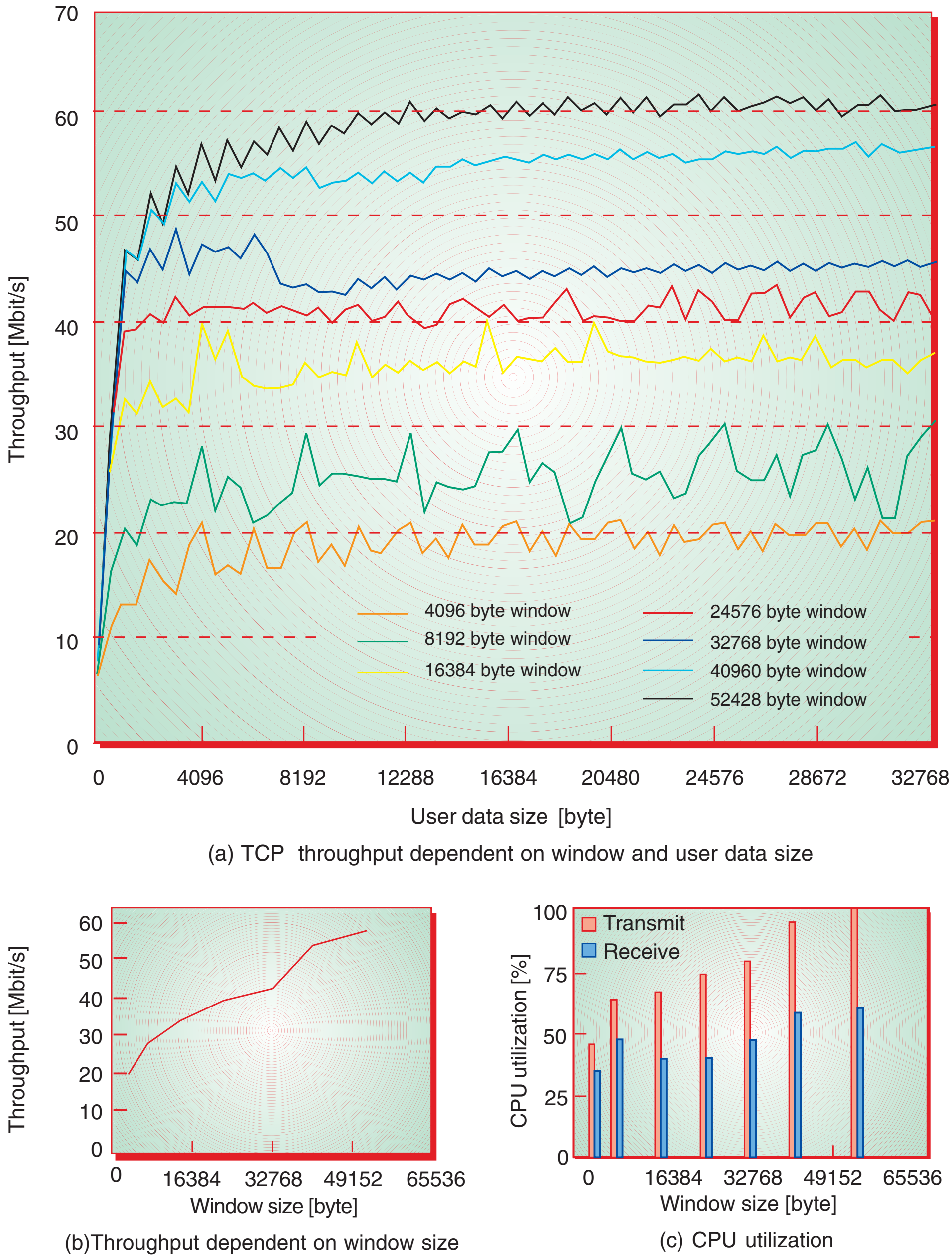


Figure 15 TCP throughput, Sparc10 SBA-200

MSS segments to be transmitted back-to-back.

There is also a throughput increase from a 40k to a 52428 byte window size. This increase is also caused by the size of the socket buffers. The segment flow is still primarily two MSSs in-between acknowledgments for these window sizes. Both window sizes have bytes for two MSS segments ready in the socket buffer when the acknowledgment arrives.

With a 52428 byte window size, the sender has buffer space for more than 5*MSS bytes. It sometimes manages to transmit 3 MSSs before the acknowledgment arrives. In short, the window is better utilized on connections with a 52428 byte window size which results in a higher throughput.

For large window sizes and large user data sizes there is no degradation in throughput as with the SBA-100 adapters. With the SBA-200 adapter, the receiver is less loaded than the sender. Thus, the read system calls return before the entire user receive buffer is filled. Thereby, the socket layer more often calls TCP to check if a window update should be returned.

7.1 Throughput peak patterns

For small window sizes there are throughput peaks which are even more characteristic than peaks observed with the SBA-100 interfaces on the Sparc10s. For example, with an 8 kbyte window the variation can be as high as 10 Mbit/s for different user data sizes. This is caused by a mismatch between user data size and window size which directly affects the segment flow. Figure 15 shows that the throughput pattern to some extent repeats for every 4096 bytes. With the SBA-200 interface, the high byte-dependent overhead of the segment processing is reduced. This implies a relative increase in the fixed overhead per segment, and the *number* of segments will have a larger impact on throughput performance. Window and user data sizes that result in the least number of segments are expected to give the highest performance. This is reflected in the throughput peaks for user data sizes of an integer multiple of 4096 bytes. With the SBA-100 adapters, the byte dependent processing was much higher, and the throughput was therefore less dependent on the *number* of transmitted segments.

For large window sizes there are no throughput peak patterns. For these window sizes, the number of outstanding unacknowledged bytes seldom reaches the window size. The segment flow is primarily maximum sized segments of 9148 bytes with an acknowledgment returned for every other segment.

8 Summary and conclusions

In this paper we have presented TCP/IP throughput measurements over a local area ATM network from FORE Systems. The purpose of these measurements was to show *how* and *why* end-system hardware and software parameters influence achievable throughput. Both the host architecture and the host network interface (driver + adapter) as well as software configurable parameters such as window and user data size affect the measured throughput.

We used small optimized probes in the network device driver to log information about the segment flow on the TCP connection. The probes parsed the TCP/IP header of outgoing and incoming segments to log the window size, the segment lengths, and sequence numbers.

Our reference architecture is two Sparc2 based machines, Sun IPXs, each equipped with a programmed I/O based ATM network interface, SBA-100 with device driver version 2.0 or 2.2.6 using ATM adaptation layer 3/4. We measured memory-to-memory throughput as a function of user data and window size. From these measurements we conclude:

- The maximum throughput is approximately 21 Mbit/s when using the 2.0 network driver version. A software upgrade of the network driver to version 2.2.6 gave a maximum throughput of approximately 26 Mbit/s.
- A window size above 32 kbytes contributes little to an increase in performance.
- Increasing the window size may not result in a throughput gain using an arbitrarily chosen user data size.
- The large ATM MTU results in TCP computing a large MSS. Because of the large MSS, for small window sizes the TCP behavior is stop-and-go within the window size.

Then, we performed the same measurements on a Sparc10 based machine, Axil 311/5.1. The MIPS rating of the Sparc10 is about 4.5 times higher than the Sparc2 MIPS rating. However, due to the machine architecture, the latency between the host CPU and the Sbus network adapter is higher on the Sparc10. We presented measurements of the send and receive path of the *network driver* which support this. For small segment sizes the *total* send and receive times are also higher on the Sparc10. From these measurements we conclude:

- Maximum throughput is approximately 34 Mbit/s.
- Increasing the window size results in higher performance.
- Access to the network adapter is a larger bottleneck than on the Sparc2.
- For small windows and user data sizes the measured throughput is actually lower than on Sparc2.

The largest throughput gain is achieved by upgrading the network adapter to the SBA-200/2.2.6. The DMA-based SBA-200 adapter relieves the host from the time-consuming access to the adapter. Thus, the host resources can be assigned to higher-level protocol and application processing. From these measurements we conclude:

- Maximum throughput is approximately 62 Mbit/s.
- Increasing the window size clearly results in higher performance.
- For small windows this configuration is more vulnerable to an inefficient segment flow, because byte dependent overhead is relatively much lower compared to the fixed segment-dependent overhead.
- Variation in throughput within one window size is the highest for small window sizes. Since primarily maximum sized segments are transmitted with large windows, the higher the window size, the lower the probability of an inefficient segment flow.

References

1 Jacobsen, V, Braden, B, Borman, D. TCP extensions for high-performance. I: *RFC 1323,* May 1992.

2 Cabrera, L P et al. User-process communication performance in networks of computers. *IEEE transactions on software engineering,* 14, 38–53, 1988.

3 Clark, D et al. An analysis of TCP processing overhead. *IEEE communications magazine,* 27, 23–29, 1989.

4 Nicholson, A et al. High speed networking at Cray Research. *ACM computer communication review,* 21 (1), 99–110, 1991.

5 Caceres, R et al. Characteristics of Wide-Area TCP/IP Conversations. I: *Proceeding of SIGCOMM '91,* Zürich, Switzerland, 101–112, 3–6 September 1991.

6 Mogul, J C. Observing TCP dynamics in real networks. I: *Proceedings of ACM SIGCOMM'92,* 305–317, Baltimore, USA, 17–20 August 1992.

7 Papadopoulos, C, Parulkar, G M. Experimental evaluation of SUNOS IPC and TCP/IP protocol implementation. *IEEE/ACM transaction on networking,* 1, 199–216, 1993.

8 Postel, J. Transmission control protocol, protocol specification. I: *RFC 793,* September 1981.

9 Lyle, J D. *SBus : information, applications and experience.* New York, Springer, 1992, ISBN 0-387-97862-3.

10 Moldeklev, K, Gunningberg, P. Deadlock situations in TCP over ATM. I: *Protocols for high-speed networks '94,* Vancouver, Canada, 219–235, August 1994.

11 Leffler, S J et al. *4.3 BSD Unix operating system.* Reading, Mass., Addison-Wesley, 1989, ISBN 0-201-06196-1.

12 Advanced Micro Devices. TAXIchip TM Integrated Circuits, transparent asynchronous transmitter/receiver interface. *Am7968/Am7969-175,* data sheet and technical manual, 1992.

13 Requirements for Internet hosts – communication layers. I: *RFC 1122,* R. Braden (ed.). October 1989.

14 Postel, J. The TCP maximum segment size and related topics. I: *RFC 879,* November 1983.

15 Jacobsen, V. Congestion avoidance and control. I: *Proceedings of ACM SIGCOMM'88,* 314–329, Palo-Alto, USA, 16–19 August 1988.

16 Nagle, J. Congestion control in TCP/IP internetworks. I: *RFC 896,* January 1984.

17 FORE Systems. *SBA-100 SBus ATM computer interface – user's manual.* 1992.

18 Cooper, E et al. Host interface design for ATM LANs. I: *Proceedings of 16th conference on local computer networks,* 247–258, October 1991.

19 FORE Systems. *200-Series ATM adapter – design and architecture.* January 1994.

Investigations on Usage Parameter Control and Connection Admission Control in the EXPLOIT testbed

BY EGIL AARSTAD, JOHANNES KROEZE, HARALD PETTERSEN AND THOMAS RENGER

Abstract

This paper investigates an implemented preventive traffic control framework for ATM which is based on a Usage Parameter Control (UPC) function and a Connection Admission Control (CAC) function. First the results of experiments with the implemented dual leaky bucket mechanism as UPC function are presented. Both traffic generators and real ATM traffic sources are used. Then the CAC function is studied by multiplexing homogeneous and heterogeneous traffic mixes. Experimental results are compared with analytical calculations and simulations. Finally, the robustness of the co-operating UPC and CAC functions is studied, which is part of ongoing work. It is concluded that the traffic control framework operates as expected, and that it is possible to guarantee network performance objectives for the admitted connections, while taking advantage of the multiplexing of connections with variable cell rates.

1 Introduction

Asynchronous Transfer Mode (ATM) is currently used in many testbeds and field trials all over the world. A lot of activities are going on to gain experience with real ATM traffic to settle yet unsolved topics. Efficient traffic and congestion control is required for ATM networks in order to support the wide variety of Quality of Service (QoS) demands for the evolving broadband services and applications. In this area further progress and experience are needed.

Many traffic experiments are carried out on the EXPLOIT Testbed (ETB) which was developed under the RACE Programme and are located in Basel, Switzerland (8). The aim of these experiments is to validate theoretical work and to gain experience on traffic performance in an ATM network. Main parts of the ETB have been developed in the previous RACE I project 1022, 'Technology for ATD'. This former testbed has been extended by interworking equipment allowing the interconnection with other RACE (e.g. BRAVE, TRIBUNE) and non-RACE (e.g. BASKOM, BATMAN, MEGACOM) broadband pilots throughout Europe. Figure 1 shows a schematic block diagram of the ETB configuration. It consists of four main subsystems, namely the switching systems LaTEX (Local and Transit Exchange) and ALEX (Alcatel Local Exchange), the RUM (Remote Unit Multiplexer), and the R-NT2 (Residential Network Termination Type 2). Many different terminals (N-ISDN, CBO-TV, VBR-MM, PC-banknote application, HQ-Audio) are connected to these subsystems via appropriate Terminal Adapters (TA). Three different types of ATM test equipment are available for traffic generation and analysis: the Alcatel 8643 ATGA, the Alcatel 8640 Test System and the Wandel & Goltermann ATM-100. They may be connected to the different subsystems in the ETB. The operation of each subsystem as well as of the whole testbed can be controlled by workstations.

A preventive traffic control framework consisting of a Usage Parameter Control

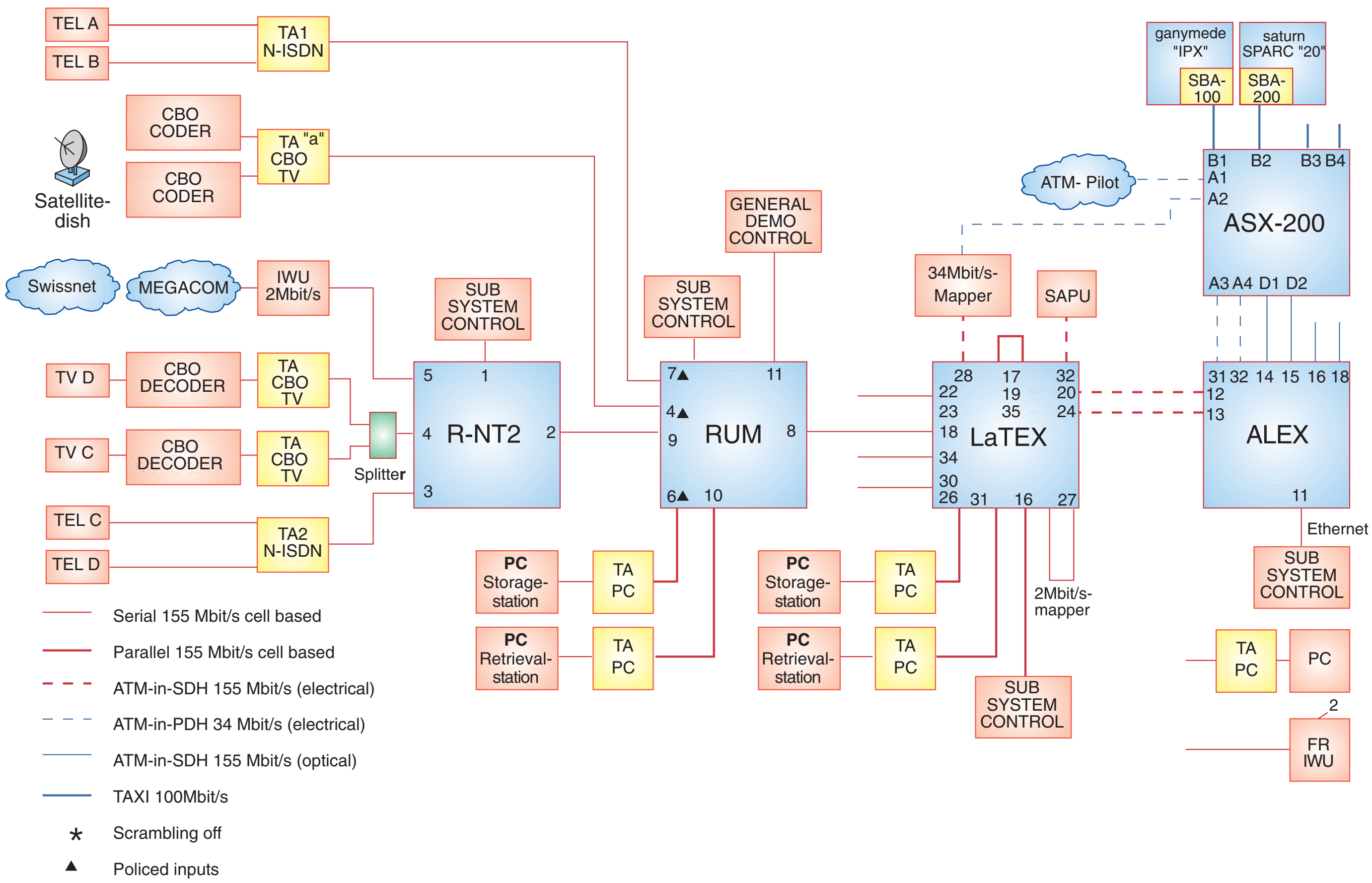


Figure 1 EXPLOIT Testbed Configuration

(UPC) function and a Connection Admission Control (CAC) function has been standardised for ATM by the ITU-T (4) and is mainly adopted by the ATM Forum (1). The control functions are based on a description of the traffic characteristics of each connection by a set of traffic parameters, for which the Peak Cell Rate (PCR) and the Sustainable Cell Rate (SCR) have been standardised.

The UPC function is required at the access point to the network to ensure that the traffic streams entering the network are within their traffic contracts. The UPC function should detect non-conforming ATM cells and take appropriate actions to minimise the potentially negative effects of malicious or unintended excess traffic. The continuous-state leaky bucket is one version of the reference algorithm which determines whether a cell is conforming with the traffic contract or not. The Peak Cell Rate of an ATM connection is defined as the inverse of the minimum inter-arrival time between two ATM cells (at the physical layer service access point). Necessary tolerances in the PCR control to cope with Cell Delay Variation (CDV) have been introduced by specifying a maximum acceptable CDV, denoted as CDV tolerance τ. The Generic Cell Rate Algorithm (GCRA) specified by the ATM Forum (1) leads to the same definition of conformance. This algorithm is also used to control the SCR by applying different parameters. The source may send continuously at the sustainable cell rate, however bursts may be sent at the peak cell rate instead. The Burst Tolerance (BT) parameter in the SCR control limits the Maximum Burst Size (MBS) which is defined by the maximum number of cells that may pass the UPC at PCR. The intention is to obtain statistical multiplexing gain, when SCR is less than PCR, by admitting more connections than when only PCR is used for resource allocation.

The CAC function is invoked for each connection request to check whether the new connection may be established between source and destination without degrading the network performance of any connection sharing the same network resources. The CAC function should be robust with respect to unexpected traffic fluctuations and conservative with respect to the admissible load.

In recent years various traffic control schemes have been studied analytically and by simulation, but a final validation can only be given by experiments and measurements within a real ATM network. Especially the control of real applications and services with yet unknown traffic characteristics is of interest. The ETB offers a great opportunity to solve some open problems by performing experiments in an environment as close to real life as possible.

This paper is organised as follows: Section 2 presents the results on the UPC function, for which deterministic, non-deterministic as well as real ATM traffic from a digital video connection have been used. Section 3 is devoted to the CAC function. Experimental results on multiplexing of homogeneous traffic mixes together with the admission boundaries for heterogeneous traffic mixes are reported. The measurements are compared with simulations and analytical results. Section 4 addresses the robustness of the total framework of the co-operating UPC and CAC functions. Here, initial results of experiments are presented where influences of burst level violations of the traffic contract have been studied. Finally, Section 5 presents conclusions and an outlook on future work.

2 Experiments on Usage Parameter Control

The objective of the UPC experiments is to verify that the implementation of the UPC function performs its task accurately and with sufficient flexibility to cope with real applications. As a first step we want to show that the UPC implementation discards cells which do not obey the traffic contract. This is checked in Subsection 2.1 for the PCR with a deterministic source with a constant cell rate. Enforcement of the SCR is verified in Subsection 2.2 with two types of non-deterministic ON/OFF sources. Finally, we show in Subsection 2.3 that suitable traffic contract parameters can be found for a digital video source as an example of a real traffic application.

Figure 2 shows the set-up for the UPC experiments. The traffic source under study is multiplexed with background traffic in order to impose additional CDV. By changing the load or the profile of the background traffic it is possible to study the influence on the foreground traffic. After leaving the multiplexer, the traffic enters the UPC function, where only compliant cells are passed on to the analyser.

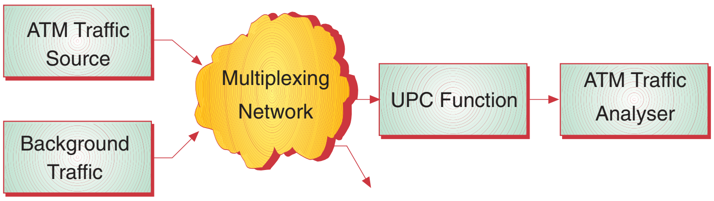


Figure 2 Configuration for the UPC experiments

2.1 Enforcement of the Peak Cell Rate

The traffic contract for the control of the PCR comprises both the contracted PCR and the CDV tolerance τ. In practice, every cell stream suffers from some CDV caused by multiplexing of cell streams or by the use of a slotted transmission medium. Some early experiments have clearly shown the influence of these causes on the CDV (2), (9). Furthermore, the influences of the traffic load of the multiplexer and of the profile of the background traffic were shown.

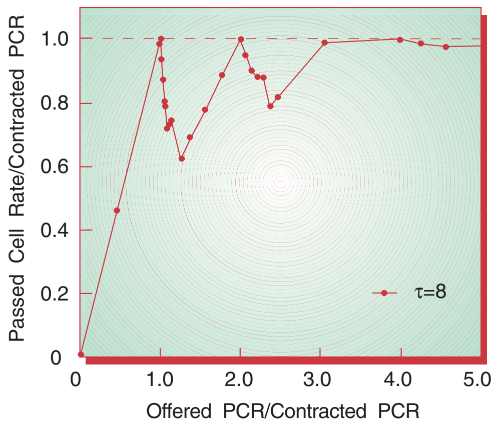


Figure 3 PCR throughput characteristics

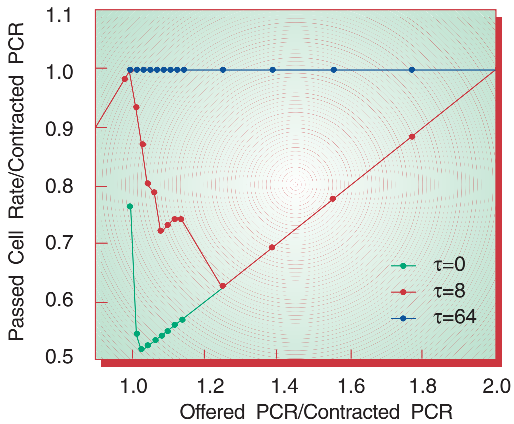


Figure 4 Magnified view of the dip

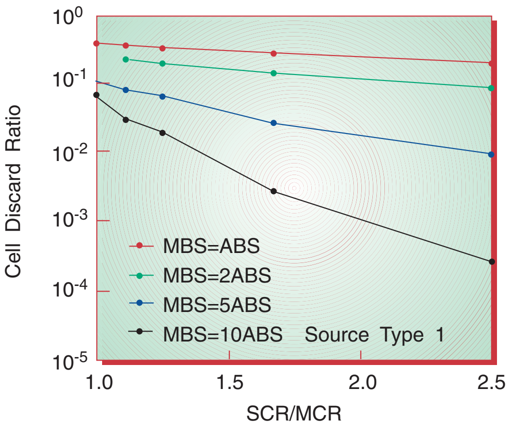


Figure 5 CDR for traffic type 1

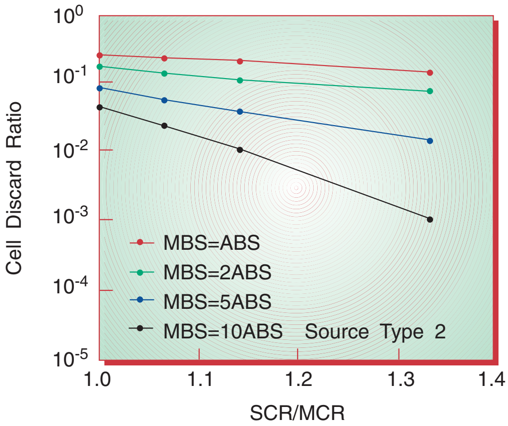


Figure 6 CDR for traffic type 2

We have gained experience on how to set the CDV tolerance for complying sources with these experiments.

The discard behaviour of the implemented UPC function is studied by measuring the UPC throughput as a function of the PCR of the offered traffic, keeping the contracted PCR fixed at a gross 2.44 Mbit/s. The measurements were repeated for three values of the CDV tolerance, chosen to be 0, 8 and 64 slots on a 155.52 Mbit/s link. It is clear from Figures 3 and 4 that the throughput depends both on the offered PCR and on the CDV tolerance. As long as the actual PCR does not exceed the contracted PCR, all cells pass the UPC. A source which is sending traffic with a PCR just above the contracted rate, can have less cells passed than a source which is violating the traffic contract by a factor of two (or more). This is shown more clearly in Figure 4 which covers the region of small contract violations. The dips in the throughput curve vanish when the CDV tolerance is increased to the contracted inter arrival time (cf. $\tau = 64$). All these results are as expected from theoretical considerations.

2.2 Enforcement of the Sustainable Cell Rate

For sources sending with variable rates we may enforce both the PCR and the SCR to be able to achieve statistical multiplexing gain. This is realised using a Dual Leaky Bucket (DLB) mechanism with two leaky buckets in parallel. The traffic contract for SCR comprises both the contracted SCR and the burst tolerance τ_s, which allows bursts at cell rates above the SCR. With some initial experiments SCR control was validated with periodic deterministic cell sequences (2), (9). Here we present the results for SCR control of non-deterministic traffic generated by an ON/OFF source with two states. If the source is in the ON state it sends at a constant cell rate and no cells are sent if the source is in the OFF state. The sojourn time distributions of both states can be negative exponential or Erlang-*k*. The various parameters of the three different types of ON/OFF traffic sources used in this paper are summarised in Table 1.

In a first step we focus on establishing values for the SCR and for the burst tolerance such that for a complying source not too many cells are discarded. We have measured the Cell Discard Ratio (CDR) as a function of the SCR for four values of the burst tolerance. In Figures 5 and 6 the SCR has been normalised by the Mean Cell Rate (MCR) of the source. The burst tolerance is expressed as the Maximum Burst Size (MBS) in terms of the Average Burst Size (ABS). The curves reveal that the CDR decreases approximately exponentially when the SCR is increased and MBS is kept constant. The slope of the curves and the CDR value where the SCR equals the MCR is determined by the MBS parameter. Furthermore, the various curves show that even with a relatively high SCR and a large burst tolerance the discard ratio does not drop down to acceptable values, which are considered to be in the range of 10^{-10}. The measurement results are as expected from a fluid flow approximation (11). Additional experiments (10) confirm that the dimensioning of SCR and burst tolerance can be more tight to the average parameter values of the source if the state sojourn times are not taken from a negative exponential distribution, which suffers from a fat tail. This can be interpreted as a requirement for applications to limit the burst size if statistical multiplexing gain is to be achieved.

2.3 Digital video as a real application

Besides artificial traffic generated by ATM traffic generators we also experimented with real traffic from the various terminal equipment and applications available at the ETB. As an example we present here the results on enforcing the PCR of digital video traffic. Forward Error Correction (FEC) combined with bit or byte interleaving is applied to the digitised video signal, before it is inserted into ATM cells using AAL1 (3). This results in an ATM physical rate of 41.366 Mbit/s. In Figure 7 the CDR at the UPC is plotted as a function of the CDV tolerance τ. Each curve corresponds with a value of the contracted PCR, which is expressed in terms of the PCR of the source. Whenever the actual PCR exceeds the contracted PCR (i.e. $p < 1$), changing the CDV tolerance has only little influence on the CDR, see the upper right part of Figure 7. Once the CDV tolerance is larger than the inverse of the contracted PCR, the UPC discards exactly the excess amount of traffic. If the actual PCR of the source is below the contracted PCR (see the lower left part of Figure 7), discards are only caused by cell clumping. Increasing the CDV tolerance drastically reduces the CDR to an acceptable level.

Table 1 Characteristic parameters of the ON/OFF traffic types

Traffic Type	Cell Rates [cells per slot]		Gross Bit Rates [Mbit/s]		Avg. Burst Size [cells]	State Sojourn Time Distribution	Burstiness
	Peak	Mean	Peak	Mean			
1	1/5	1/25	31.10	6.22	1467.2	neg.-exp.	5
2	1/80	1/160	1.94	0.97	229.25	neg.-exp	2
3	1/20	1/400	7.78	0.39	183.4	Erlang-10	20

While enforcing the PCR, the resulting picture quality was assessed by a small group of people in terms of flicker, loss of picture, etc. We discovered that besides the value of the cell loss ratio, also the structure of the cell loss process (e.g. correlated cell losses) has a strong influence on the picture quality. However, the current definition of network performance in (5) does not include parameters to describe this structure of the cell loss process.

3 Experiments on Connection Admission Control

The aim of the CAC experiments is to investigate the ability of different CAC algorithms to maintain network performance objectives while exploiting the achievable multiplexing gain as much as possible. We focus on the Cell Loss Ratio (CLR) (defined in (5)) as network performance parameter in our studies. Our performance study of various CAC algorithms is not restricted to the two-level CAC implementation in the ETB (see (7) for a description).

In a first step we measure the CLR at a multiplexer which is fed by several traffic sources. By varying the number and the type of the sources we can determine the admissible number of sources for a given CLR objective. The set of these points is referred to as the admission boundary. Once we have a measured admission boundary it can be compared with the corresponding boundaries obtained when applying a certain CAC mechanism. If the CAC boundary is above the measured boundary, the investigated CAC performs a too optimistic bandwidth allocation and is not able to maintain network performance objectives. On the other hand, if the CAC boundary is below the measured boundary the available resources may not be optimally used by the CAC mechanism. We have selected CAC algorithms based on a bufferless model of the multiplexer, since the small buffers in the ETB switches can only absorb cell level congestion. In this paper we only consider the well-known convolution approach. Results for other mechanisms can be found in a more detailed report (10).

Figure 8 depicts the configuration. The cells generated by *N* sources are all multiplexed in the multiplexer under test, but first the cell streams pass another multiplexing network where the characteristics can be modified. It is ensured that cells are only lost by buffer overflow at the multiplexer under test. Since the number of currently available real sources and applications in the ETB is limited, we have used ATM test equipment for the generation of statistical traffic with various profiles. Multiplexing has been performed at an output link where a buffer size of 48 cells and an output capacity of 155.52 Mbit/s are available. The CLR has been measured for the aggregate traffic. This is taken into account for the convolution algorithm by applying the global information loss criterion (6) for the calculation of the expected overall cell loss probability.

3.1 Homogeneous traffic mix

First the case of a homogeneous traffic mix is considered where all sources have identical characteristics. The same ON/OFF sources as for the UPC experiments have been used (see Table 1). Figures 9 and 10 show the measured cell loss ratio as a function of the number of multiplexed sources. Simulation results with 95 % confidence intervals and analytical results from the convolution algorithm are included in the figures. When comparing measurement, simulation and convolution results, the convolution gives the largest cell loss ratio, while the measurement shows the smallest loss values. This is expected, since the convolution is based on a bufferless model of the multiplexer such that buffering of cells at the burst level is neglected. The very good coincidence of measurement and simulation is remarkable. The admissible number of sources for a given CLR objective can be easily obtained from the figures. For traffic type 1 almost no multiplexing gain is achievable while the lower peak cell rate of traffic type 2 allows a reasonable gain.

3.2 Heterogeneous traffic mix

Now sources from both traffic types are multiplexed and the resulting CLR is measured. The following definition has been used to determine the set of admission boundary points: for each boundary point adding only one further source of any traffic type would already increase the resulting CLR above the objective. Figures 11 and 12 show the measured admission boundaries for CLR objectives of 10^{-4} and 10^{-6}, together with analytic results for the convolution algorithm and PCR allocation. The results reveal that multiplexing gain can only be achieved

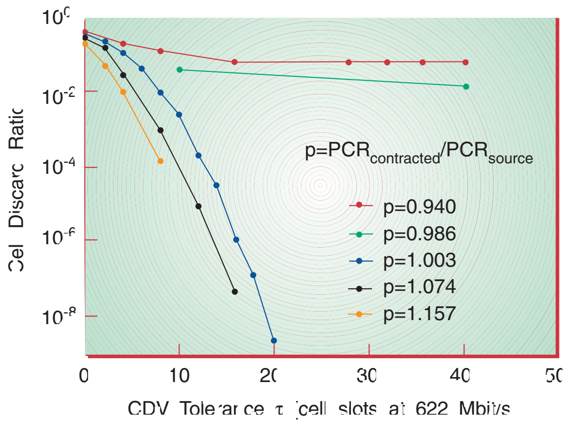


Figure 7 CDR for video traffic

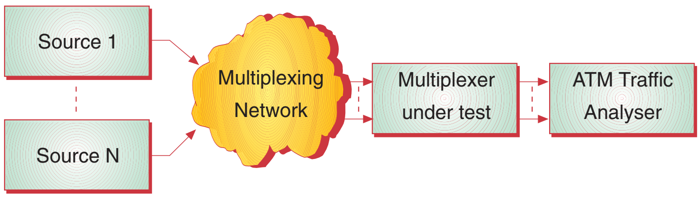


Figure 8 Configuration for the CAC experiments

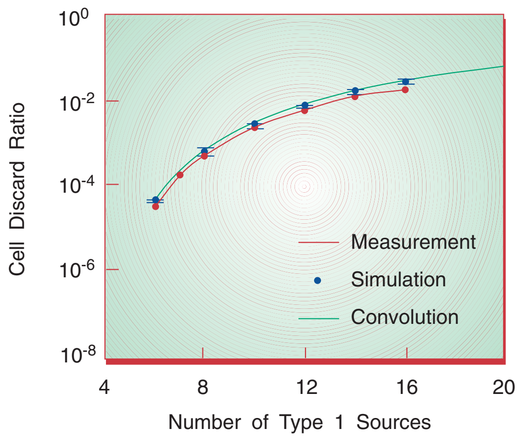


Figure 9 CLR at the multiplexer, type 1

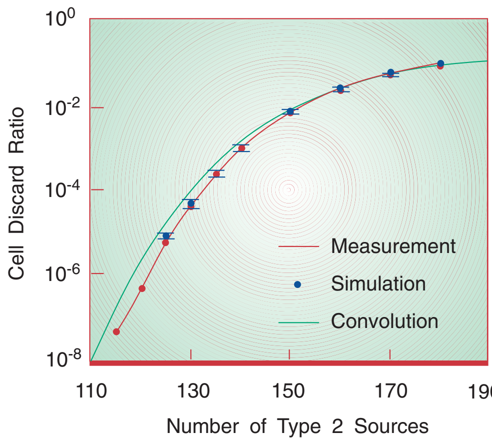


Figure 10 CLR at the multiplexer, type 2

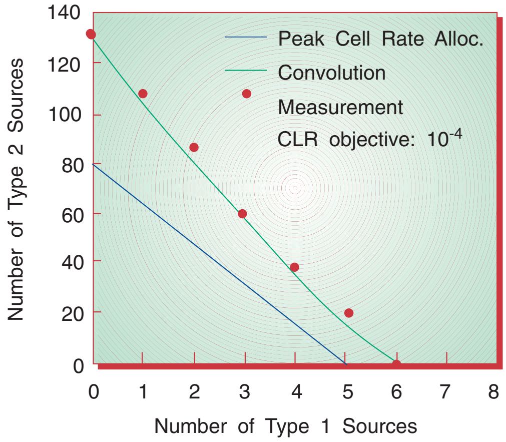


Figure 11 Boundaries for CLR = 10^{-4}

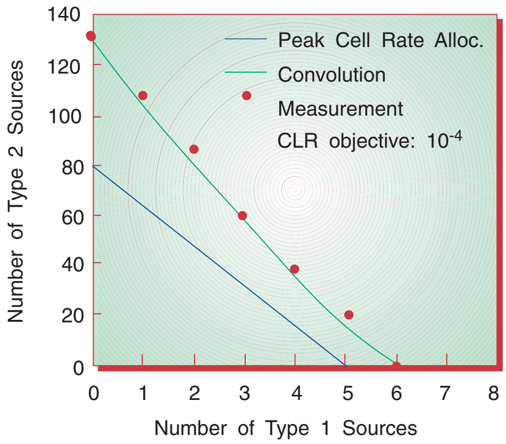


Figure 12 Boundaries for CLR = 10^{-6}

when a significant portion of the traffic consists of type 2 sources. It is also evident that a smaller CLR objective reduces the achievable multiplexing gain. The comparison of the convolution boundaries with the boundary points found by measurement confirms the ability of the CAC scheme to maintain CLR objectives while allowing to exploit a significant part of the achievable multiplexing gain. It must be noted, however, that the source characteristics were not yet changed to stress the multiplexer, which is investigated in the next section.

4 Experiments on robustness of the UPC and CAC framework

While UPC and CAC functions have been studied separately in the previous sections, we now focus on their co-operation. The aim is to identify whether this traffic control framework is able to maintain network performance guarantees under various network conditions and load scenarios.

The experiment configuration is almost the same as for the CAC experiments (see Figure 8). The major difference is that all traffic now passes the UPC function before being multiplexed. For each source a traffic contract based on its statistical parameters is established and enforced during the measurement. The number of established connections is determined by the CAC function applying the convolution algorithm. The convolution is based on the two UPC parameters, PCR and SCR, and not on the statistical parameters of the sources. While for the PCR there is no difference between these two methods describing a source, the SCR is always larger than the Mean Cell Rate (MCR). The ratio SCR/MCR depends on the characteristics of the source and can be rather large. The used configuration allows to measure the cell loss ratios at both the UPC (called Cell Discard Ratio, CDR) and at the multiplexer (Cell Multiplexing loss Ratio, CMR). The CDR and the CMR can be combined to obtain the overall Cell Loss Ratio (CLR) for each traffic source. We investigate traffic mixes on the admission boundary such that the highest possible values for the CMR may be achieved, since the CMR increases monotonously with the number of multiplexed sources.

The robustness of the traffic control framework is now investigated by introducing different kinds of changes in the characteristics of a major part of the sources. However, the remaining sources will not be changed to be able to see if their perceived network performance parameters are influenced by the other traffic.

In a first series of measurements we have investigated the robustness with respect to changes in the mean cell rate. For these experiments we have used sources according to traffic type 3 (see Table 1). These sources still have an ON/OFF characteristic but state sojourn times are distributed according to an Erlang-10 distribution in order to be able to fix the SCR closer to the MCR. The UPC parameters in the traffic contract have been chosen such that we can expect a CDR smaller than 10^{-5} for a compliant source (i.e. source with parameters according to Table 1). We call a source non-compliant if the mean cell rate is increased either by decreasing the OFF-state duration or by increasing the ON-state duration. Approximately 1/3 of the sources have been chosen as well-behaving, and the rest equally split as misbehaving in the two different ways. For the CAC a CMR target value of 10^{-4} has been taken.

The results of this experiment are depicted in Figures 13 and 14. Each figure shows the various loss ratios as a function of the mean cell rate of the non-compliant sources normalised to the SCR contained in the traffic contract. Figure 13 clearly shows that compliant sources are protected since the overall loss ratio expressed by CLR remains rather unchanged for these sources, even if the mean of the non-compliant sources is nearly doubled. In all cases the value of CLR is below the target value of 10^{-4}. However, sources violating the traffic contract suffer from cell discards at the UPC (see Figure 14). Both figures also reflect the fact that the overall loss ratio at the multiplexer is hardly influenced at all by traffic contract violations since the excess traffic is already discarded at the UPC. Furthermore, since these discards at the UPC seem to smooth the output traffic before this is multiplexed, the CMR does not increase even if the aggregate load of the multiplexer increases by approximately the factor SCR/MCR for the case with the highest mean rates, where MCR refers to the rate of the compliant sources.

In another series of experiments the traffic characteristics of the sources are modified by cell level changes to investigate the impacts on the network performance. The CDV tolerance τ in the PCR control is increased, thereby allowing an increasing number of back-to-back cells to pass the UPC and enter the multiplexer. With these experiments we study the sensitivity of the network performance with respect to sources sending clumps of back-to-back cells. This sensitivity may be crucial because the values for the CDV tolerance τ are not used by the investigated CAC algorithms. Bounds on acceptable values for τ may be needed to ensure the robustness of the framework. Some initial results were already obtained, but further investigations in this area are necessary before conclusions can be drawn.

5 Conclusions and future work

Experimental results obtained in the EXPLOIT Testbed on Usage Parameter Control (UPC) and Connection Admission Control (CAC) have been presented and compared with analytical calculations and simulations.

With the UPC experiments the implemented Dual Leaky Bucket (DLB) mechanism was validated to operate correctly by enforcing both the Peak Cell Rate (PCR) and the Sustainable Cell Rate (SCR). As expected, the SCR and the Maximum Burst Size (MBS) cannot always be set very tight to the average source characteristics, since that would cause an unacceptable high cell discard ratio for complying sources with exponential sojourn time distribution. Tighter values may be chosen if the probability of long bursts is reduced. When designing protocols to communicate over ATM networks such properties should be taken into account.

The performance of several CAC algorithms has been evaluated. The ability of these CAC mechanisms to control the network performance (cell loss ratio) was investigated by multiplexing both homogeneous and heterogeneous traffic mixes with various non-deterministic ON/OFF sources. The convolution algorithm always performs a conservative allocation of resources, but the admitted number of sources is rather close to the maximum possible, given the small buffer sizes implemented in the switches of the testbed.

The robustness of the co-operating UPC and CAC functions has been partly examined. In the cases considered so far, the well-behaving sources are protected even if approximately 2/3 of the sources are misbehaving by increasing their mean load.

In the near future our investigations will include other types of violating sources and more real sources having variable cell rate characteristics, like multi media applications or traffic originating from LAN environments. Furthermore, the excellent interconnection possibilities of the EXPLOIT Testbed will be utilised to validate the traffic control framework with long distance connections using the ATM Pilot network.

Acknowledgements

This work has been carried out as part of the RACE II project EXPLOIT. The authors would like to thank all the people who contributed to EXPLOIT and who assisted and supported us on-site. Special thanks to A. Bohn Nielsen, H. Hemmer, J. Witters and R. Elvang who made significant contributions to obtain the presented results.

Abbreviations

ABS	Average Burst Size
BT	Burst Tolerance
CAC	Connection Admission Control
CDR	Cell Discard Ratio
CDV	Cell Delay Variation
CLR	Cell Loss Ratio
CMR	Cell Multiplexing loss Ratio
DLB	Dual Leaky Bucket
ETB	EXPLOIT TestBed
FEC	Forward Error Correction
GCRA	Generic Cell Rate Algorithm
MBS	Maximum Burst Size
MCR	Mean Cell Rate
PCR	Peak Cell Rate
SCR	Sustainable Cell Rate
UPC	Usage Parameter Control

References

1 ATM Forum. *ATM user-network interface specification,* Version 3.0, September 1993.

2 Bohn Nielsen, A et al. First results of UPC experiments in the EXPLOIT testbed. I: *Proceedings of the EXPLOIT Traffic Workshop,* Basel, 14–15 September 1994.

3 Hessenmuller, H, Nunes, S. A terminal adapter for high-quality audio and video signals to be used in a B-ISDN based on ATD. I: *Proceedings of the International Symposium on Broadcasting Technology,* Beijing, September 1991.

4 ITU-T. *Traffic control and congestion control in B-ISDN. Recommendation I.371,* Frozen Issue, Paris, March 1995.

5 ITU-T. *B-ISDN ATM cell transfer performance.* Draft recommendation I.356, Geneva, January 1993.

6 Jabbari, B, Yegenoglu, F. An efficient method for computing cell loss probability for heterogeneous bursty traffic in ATM networks. *International journal of digital and analog communications systems,* 5, 1992, 39–48.

7 Mitrou, N M et al. Statistical multiplexing, bandwidth allocation strategies and connection admission control in ATM networks. *European transactions on communications,* 5, March/April, 1994, 33–47.

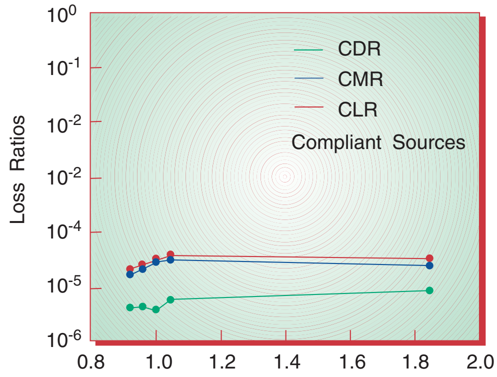


Figure 13 Loss ratios for original sources

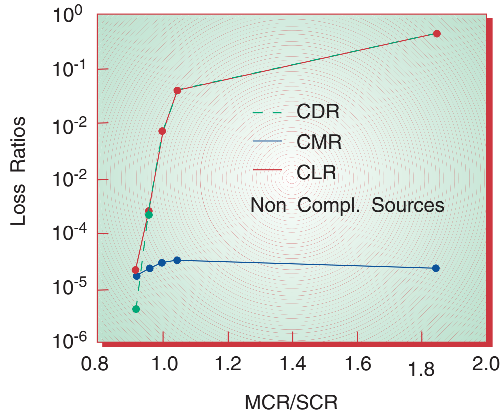


Figure 14 Loss ratios for misbehaving sources

8 Potts, M. EXPLOITation of an ATM testbed for broadband experiments and applications. *Electronics & communication engineering journal,* December 1992, 385–393.

9 RACE 2061 WP3.1. *Results of experiments on traffic control using test equipment.* RACE 2061 Deliverable 18, June 1994.

10 RACE 2061 WP3.1. *Results of experiments on traffic control using real applications.* RACE 2061 Deliverable 28, December 1994.

11 Yin, N, Hluchyi, M G. Analysis of the leaky bucket algorithm for on-off data sources. *Journal of high speed networks 2,* IOS Press, 1993, 81–98.

Synthetic load generation for ATM traffic measurements

BY BJARNE E HELVIK

Abstract

The only way to validate the performance of an ATM based network is extensive traffic measurements. Obviously, if such measurements shall be meaningful, they must be carried out under a load with characteristics similar to that of real traffic. The network performance measures obtained, e.g. cell loss, delay and jitter, will depend heavily on the load put on the system. The paper starts with a brief discussion of this issue. Next, an introduction to modelling of ATM traffic sources is given. Various approaches to traffic generation is outlined, before the modelling framework on which the STG is based, is presented. The STG (Synthesised Traffic Generator) is an ATM traffic generator, which is designed to generate a traffic load as realistic as possible, and thereby enable trustworthy ATM system measurements. The source type models include the behaviour of physical sources and the formation of ATM cell streams done by the terminal equipment. How such source type models may be established is discussed and exemplified. In the STG, time varying traffic mixes towards various destinations may be defined. These are denoted load scenarios. One STG can generate the multiplexed traffic stemming from up to 2047 independent sources. A nearly infinite number of independent and reproducible cell load sequences, with the same statistical properties, may be generated for each scenario. The paper concludes with a sketch of how the STG functionality may be extended.

1 Introduction

Today, there are few who doubt that the coming generation of high capacity networks and switching systems, both in the public and private domain, will be based on the ATM (Asynchronous Transfer Mode) principle. In the public domain switches from various manufacturers are tested in trial networks and "ATM solutions" are heavily marketed for local area networks. However, ATM is new and unproven – also with respect to traffic engineering. The capacity (bandwidth) on demand idea underlying ATM poses a wide range of challenges. The characteristics of the load put on the systems are partly unknown and new call handling and switching techniques are introduced. At the same time very strict requirements are put on network performance/quality of service parameters like cell loss frequencies and cell delays. In spite of the enormous effort put into traffic research related to ATM during the last years, our possibilities to perform a thorough validation of the traffic carrying capabilities of these new systems is restricted.

- A sufficiently detailed *mathematical analysis* under realistic assumptions is impossible, even for small systems.
- Full scale *simulations* are too demanding with respect to computer time. This is, to a large extent, caused by the enormous number of cells that pass through the system relative to the number of events occurring during the same time. Work is going on toward special "speed-up techniques", see [1]. However, this evaluation technology is not yet mature.
- *Measurements* and experimentation then remain as our prime means to gain confidence in this new and not yet fully proven transfer technology. Types of measurements that will be carried out are for instance:
  - Validation of equipment, like the cell carrying capability of a switch, the jitter it introduces in a cell stream, its fairness with respect to service impairments caused to various connections, etc.
  - Design decisions and trade-offs, like determining the appropriate size of packets on various protocol levels, the (possible) forward error correction (FEC) needed for some real time services, etc.
  - QoS cross-influence between services (source-types). For instance, is there a difference in the service impairments when a high quality video connection competes in the network with mainly data connections, constant bitrate connections (CBR) or other video connections when the average load is the same.
  - Investigation of acceptable loading under various traffic mixes to validate connection acceptance control (CAC) functionality.

When carrying out measurements, it is necessary to be able to register for instance the delay of various information units (cells, transport packets, video frames, etc.) through the network as well as the cell loss associated with the same units. Note, however, that *the traffic load put on the system is the most important constituent with respect to the results obtained*. The best approach is of course to put controlled, real traffic on the system. This, however, is not feasible. Real traffic can not be kept sufficiently stable and be varied with respect to load and mix to obtain controlled and accurate results. A realistic loading of a system requires an array of independent generators, each producing the traffic stemming from a large number of independent sources. In addition to established sources/services, it should be possible to load ATM systems with traffic from foreseen, but not yet available services, and traffic corresponding to various choices with respect to service specifications and terminal equipment design.

High level requirements for an ATM traffic load generator

- The traffic load should stem from a large number of independent sources (at least 1000), each with its unique virtual circuit (VC), payload, etc.
- The models of the sources should be physically interpretable, reflect both the information transfer caused by the application/service, and the protocols and equipment of the end systems. The model(s) should be adaptable to a wide range of source types, e.g. speech, variable bitrate coded video, various kinds of data traffic, and new and yet unknown services/equipment.
- The generated traffic must reproduce the short and long term properties of real traffic, i.e. the range from microseconds to hours.
- It should be possible to define load scenarios with a mix of traffic from many different types of sources with different and time-varying traffic interests.
- A nearly infinite number of independent and reproducible generations of each scenario should be possible.
- The underlying traffic model(s) should enable a compact and efficient hardware implementation.

This in order to test the effect of these on the network performance and the QoS before decisions are made. Hence, in order to perform traffic measurements of ATM systems, it is necessary to be able to generate an artificial traffic load which has characteristics which are close to real (current and expected) traffic. For details, see the separate box *High level requirements for an ATM traffic load generator*.

The recognition of the above facts triggered our work on source modelling and traffic generation. Traffic generation must be based on one or more models of the source. Classes of models and generation principles are discussed in the next section, together with the composite source model (CSM) on which the synthesised traffic generator (STG) is based. This generator, which currently is the most advanced ATM traffic generator available, is presented in Section 4. Before that, Section 3 gives a brief introduction to how source type models may be defined in the composite source model framework. An outlook and some concluding remarks end the paper.

2 Models for load generation

2.1 Source type modelling

Before proceeding to modelling of sources for load generation, a brief overview of different types of source models is necessary. The subsections below are neither claimed to be a complete review of the state of the art, nor perfect in their classification.

2.1.1 State based models

The most common approach to ATM source type modelling is to assume that there is some sort of finite state machine (FSM) behaviour determining the behaviour of the source. For some types of sources, for instance for speech encoded with silence detection, this is a correct assumption. For other source types, for instance variable bitrate coded (VBR) video, it is an artefact used to approximate an infinite state behaviour of the source. This will be dealt with in Section 3.

There are two different approaches in state based modelling. Either the cell generation process is modelled by a finite state machine directly, or the cell generation process is modulated by a finite state machine.

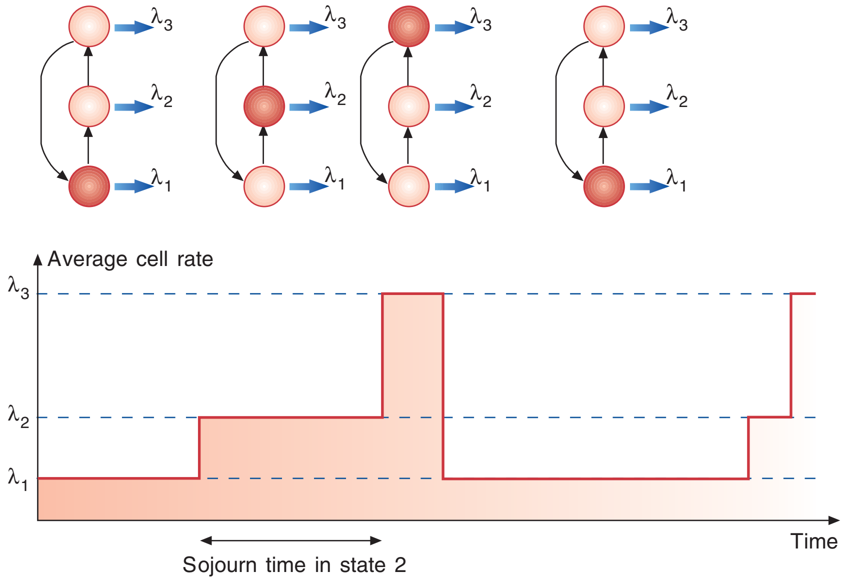


Figure 2.1 Illustration of a simple modulating finite state machine

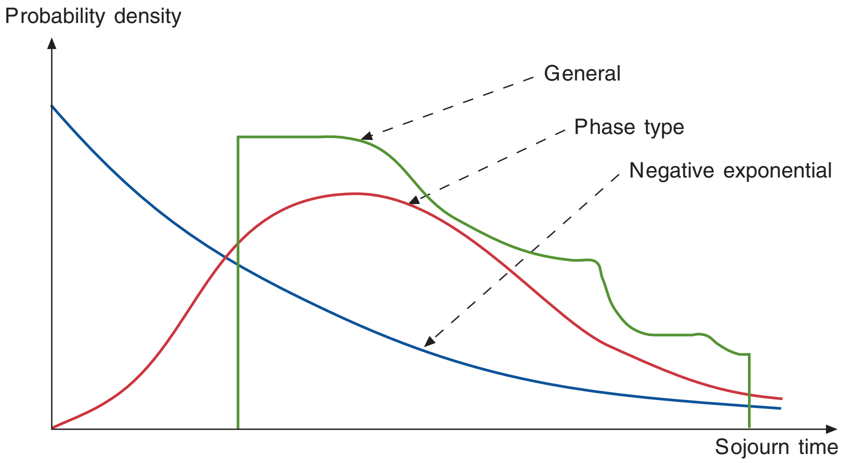


Figure 2.2 Sketch of different types of sojourn time distributions

Modulation of the cell process

This approach reflects the multilevel modelling which will be discussed later in this section. The basic idea is that a finite state machine modulates the expected rate (mean value) of an underlying process. This is demonstrated by a very simple example in Figure 2.1.

The state machine is usually assumed to have the Markov or semi-Markov property, i.e. its sojourn times in the states and the transitions to the next states are completely independent of its previous behaviour. The source type models of this class differ in the following aspects:

Sojourn time distribution; determines the duration of the levels in the modulated flow. The sojourn times are in most models negative exponentially distributed which makes the state machine a discrete state continuous time Markov model. It is also proposed to use general distributions, which makes it a semi-Markov model. As a trade-off between modelling power/accuracy and mathematical tractability, phase type distributions are suited. See [2] for an introduction. This type of sojourn time is used in the source type model used in the STG, cf. Section 2.4.1. The different types are illustrated in Figure 2.2.

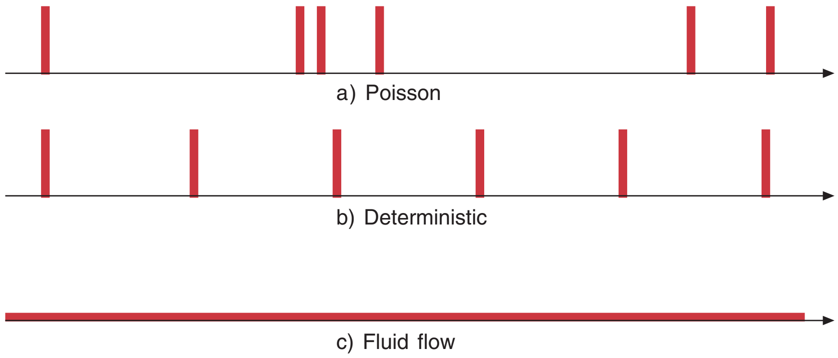


Figure 2.3 Sketch of different types of cell level processes

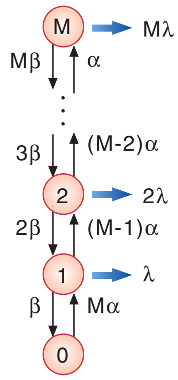


Figure 2.4 Diagram of a multi-mini-source model corresponding to M independent ON-OFF sources

The type of the modulated process; which are commonly one of the following three:

- *Poisson.* The modulated process is often assumed to be Poissonian[1]. This is done primarily for mathematical tractability without any basis in the behaviour or real source. When the state sojourn times are negative exponentially distributed, this kind of stochastic process is referred to as a Markov Modulated Poisson Process, MMPP. For processes with restriction on the size or structure of the state space, other acronyms may be encountered. For instance is the two state case often referred to as a switched Poisson process, SPP, which will be returned to below.
- *Deterministic,* which is usually synonymous with a constant cell interarrival, is a process which in many ways better reflects the lower level of an ATM source. This source type is less mathematically tractable than the Poisson process, but is often used for simulation, see for instance [3].
- *Fluid flow*. In this kind of model, the "flow" of cells from the source is approximated by a continuous fluid which enters the system, i.e. the distinct cells are not studied in the model. This type of models are aimed at mathematical analysis of systems, see [4] and [5] for examples.

[1] *Cells are generated independently of all previous cells at a constant rate determined by the modulator.*

The typical behaviour of these are illustrated in Figure 2.3.

Structural aspects of the model. The last attribute in which this kind of source type models differ is the size and the structure of the state space. Restrictions are often introduced to limit the number of parameters and to increase mathematical tractability. For instance, the two state switched Poisson process (SSP) has been used to fit the average, the variance and the long and short term behaviour of a measured process, see for instance [6]. Another popular two state model is the simple ON-OFF model, where no cells are formed in the off-state.

The activity of a source is often modelled as a chain, i.e. a one dimensional structure where the next state entered always is neighbour to the current. A model of this kind is the multi-mini-source model which is shown in Figure 2.4. This source may be regarded as a multiplex of *M* independent ON-OFF sources with average on and off duration of $1/\alpha$ and $1/\beta$ respectively, and with a cell rate of λ when on. In spite of its simplicity this model has been shown to reflect some of the characteristics of a variable bitrate coded video scene reasonably well [7].

This model is unable to reflect the periodicities due to frame, line and cell similarities seen in the output from a variable bitrate coded video sources. As a result of these periodicities, S-Q Li and co-workers have in some recent papers approached the ATM traffic analysis from a spectral point of view, i.e. in the frequency domain [8, 9 and 10]. In this work they have also devised a procedure for deriving continuous time Markov modulated model with certain spectral characteristics (location and width of peaks), named the circulant chain [9]. This is a symmetric circular model where only the generated cell rate differs from state to state. The model in Figure 2.1 is an example of a very simple circulant.

Modelling the cell process directly

In this approach the cell level is modelled directly. This is most commonly done by a discrete time discrete state Markov model. One time step is equal to one cell period. The Bernoulli process is the simplest of these. In the *Bernoulli pro-*

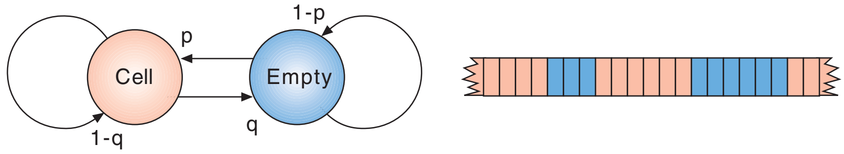


Figure 2.5 Example of a discrete time discrete state model for generation of a cell sequence
a) Simple cell generation model
b) Sample of generated cell sequence

cess a cell is arriving (being generated) in a cell slot with constant probability p. Cell arrivals/generations are independent of previous cell arrivals/generations. This corresponds to geometrically, identical and independently distributed cell inter-arrivals with mean of $1/p$ cell periods. This model is popular due to its mathematical tractability and is widely used. There is, however, no physically based motivation for why a cell stream should have this characteristic. A somewhat more sophisticated model is shown in Figure 2.5. It generates geometrically, identically and independently distributed burst of consecutive cells with mean $1/q$ and similar gaps with mean $1/p$. Also this model is rather vaguely motivated and cannot be used to properly characterise ATM source types. It is seen that it degenerates to the Bernoulli process when $1 - q = p$.

A class of processes with far better capabilities to model the characteristics of ATM traffic is the DMAP (Discrete-Time Markovian Arrival Process), introduced by Blondia [11]. However, since both cell and burst level behaviour of the source (see Section 2.3) is captured in a single level model, the number of states will become immensely large for source with a non-trivial behaviour at both layers.

A special class of source models which also should be mentioned is the *deterministic source models,* i.e. the cell from a source has always the same interarrival distance. See Figure 2.3.b. This class is used to model constant bitrate (CBR) sources, and it is the phases between the cell arrivals from the different sources which determine buffering delays and possible cell losses in the system. See for instance [12].

2.1.2 Time series

Models based on time series have been given most attention in the modelling of variable bitrate coded video sources. The information flow [bits/time unit] from a source may be modelled as a time series, where, for instance, the expected number of bits in a video frame is determined by the number of bits in the previous frames. As an example regard the following VBR video model from [13] where T_n is the number of bits per frame in the n'th frame and

$$T_n = \mathrm{Max}(0, X_n + Y_n + Z_n) \qquad (1)$$

where

$X_n = c_1 X_{n-1} + A_n$
(Long term correlation)

$Y_n = c_2 Y_{n-1} + B_n$
(Short term correlation)

$Z_n = K_n C_n$
(Scene changes)

The quantities A_n, B_n and C_n are independently and normally distributed random variables with different mean and variances, c_1 and c_2 are constants determining the long and short term correlation of the video signal and K_n is a random variable modelling the occurrence of scene changes which results in a temporarily very large bitrate. Another simple video frame oriented model, bridging the gap between the state oriented models outlined in Section 2.1.1 and the time series models, is the discrete autoregressive (DAR) model [14].

The above mentioned models assume that the cells generated from one frame are evenly distributed over the frame period and that all frames are coded similarly. Time series models including line- and frame-recorrelation from an immediate transmission of cells are proposed [15], as well as a model taking into account the lower information rate of the intermediate frames in MPEG encoded video [16].

2.1.3 Self similar stochastic processes

Rather recently, analyses of measurements of data traffic indicate that this type of traffic has self similar stochastic properties. By this is meant that the statistics of the process, e.g. the variance and autocorrelation, are the same, irrespective of the time scale regarded. The traffic process (packets per time unit) looks similar whether it is regarded per second, minute, hour, or day as illustrated in Figure 2.6. For details see [17, 18] or [19.]

Our current hypothesis is that this effect may also be captured by the multilevel activity modelling outlined in Section 2.3. Simulation and analytical studies carried out seem to confirm this hypothesis.

2.2 Three approaches to ATM load generation

Three conceptually very different approaches to the generation of ATM traffic for measurement and simulation may be identified. These are dealt with in the following subsections.

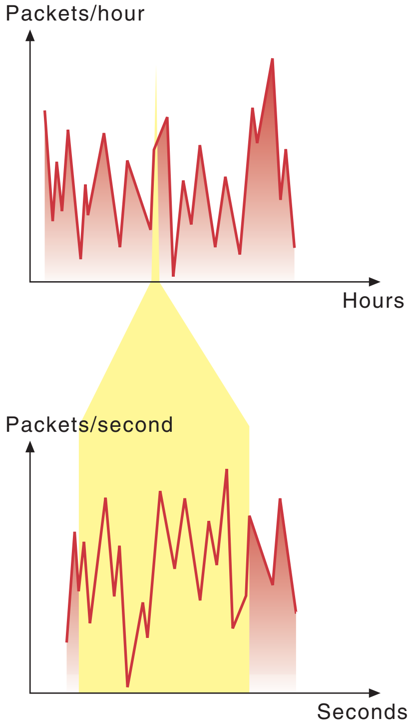


Figure 2.6 Illustration of a self similar stochastic process

2.2.1 Replay of a cell sequence

This is a *storage* based generation, where a pre-recorded or predefined cell sequence is (repeatedly) reproduced during a measurement. Due to its determinism and simplicity, this approach is suited for initial functional testing. It is, however, unsuitable for validation of traffic handling capabilities since limited length cell sequences will be available with a reasonable memory. (A complete 155 Mbit/s ATM cell stream requires 19 Mbytes/s.)

2.2.2 Generation according to a class of stochastic processes

This may be regarded as a *black box* approach. A cell stream is generated according to "some" class of stochastic process, for instance a renewal process or (multiple) on-off sources. The parameters of the stochastic process are chosen to fit some of the statistics of a source or a multiplex of sources.

In this case it is possible to generate long independent sequences, but it still remains open how well real traffic is represented. The quality of the traffic generated will depend on how well the chosen class of statistics fit the actual process. If

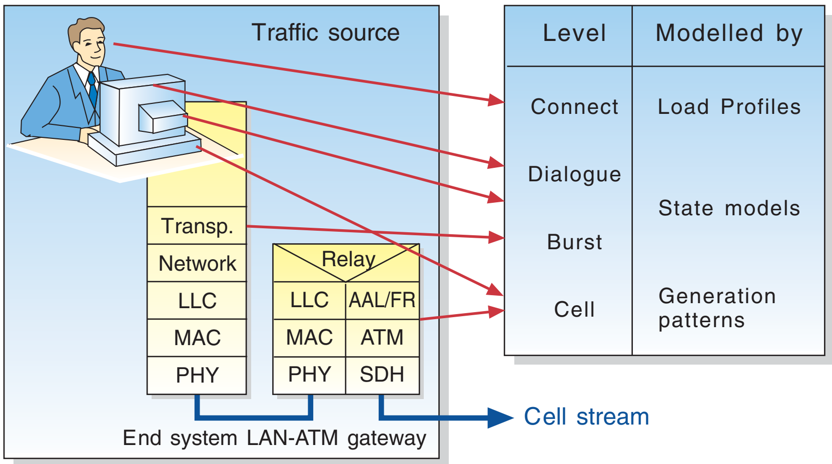


Figure 2.7 Illustration of some of the factors determining the characteristics of an ATM cell stream and the corresponding activity levels

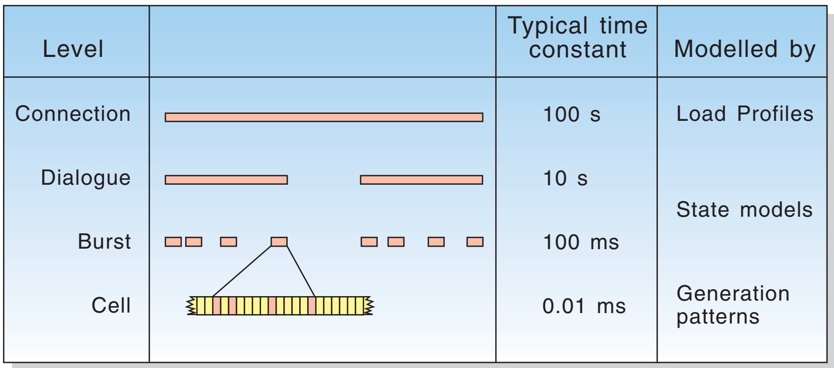


Figure 2.8 Depiction of ATM source activity and corresponding model levels

a real source is available, some of the statistics of this source may be fitted, for instance mean, variation and auto-correlation after a certain lag. There is, however, no guarantee that these are the most salient statistics with respect to traffic handling. Experimental evidence indicates the opposite [20]. If a real source from which statistics may be obtained is not available, this approach becomes inapplicable.

Traffic generation according to this principle tends to be based on simplistic models, e.g. a Bernoulli process where cells are generated with a constant probability p.

2.2.3 Physically based source models

This approach is also based on stochastic processes, but in this case the detailed behaviour of the source is attempted reproduced. It may be regarded as a *white box* approach where the inner workings of the physical source and end system are modelled together with user behaviour and other elements forming the cell stream. Our work is based on this approach. The next section discusses types of activity in a source and in Section 2.4 it is shown how these activities may be captured in a model. Section 3 discusses how this general model may be parameterised to generate realistic traffic seemingly stemming from many sources of different types.

In addition to be able to generate traffic similar to that from established sources/services, it should be noted, that by this approach it is possible to load ATM systems with

- traffic from foreseen not yet available services, and
- traffic corresponding to various choices with respect to service specifications and terminal equipment design.

This enables testing of the effect of new services, draft protocol specifications and design decisions before these are made final.

2.3 Activities in a source

The cell stream from an ATM source is determined by a number of factors. Among these are:

- The human(s) initiating and controlling the communication process, or eventually defining criteria and procedures for when and how it shall take place.
- The information to be transferred. This may in itself be time varying, e.g. a variable bitrate coded video signal.
- The speed of the equipment used in the communication, e.g. disks, transmission lines and processors.
- The protocols on various levels forming the cell stream.
- The ATM network by its influence on the end-to-end (protocol) activity.

This is illustrated in Figure 2.7, where the so-called activity levels are introduced. These are discussed below. See Figure 2.8 for an illustration. The model elements used in the composite source model to capture these activities are indicated as well. These will be described in Section 2.4.

Connection level, describing connects and disconnects of the virtual circuit. Typical time constants are in the order of 100 s. This level is related to the command function associated by the physical source. When to communicate (the time when a call is established) is in the STG modelled by the load profile, see Section 2.5. Where to communicate (call destination) is modelled by the cell header (VCI, VPI, destination).

Dialogue level, describing the alternations between the A and B party in a connection. Typical time constants are in the order of 10 s.

Burst level, describing the behaviour of the active party. The activities on this level are primarily governed by the physical source. Time constants are in the order of 100 ms. The dialogue- and burst levels are modelled by a finite state machine. See Section 2.4.1.

Cell level, describing the actual cell generation. This level is primarily governed by the terminal and/or end system. Time constants are in the order of 0.01 ms. The cell level is encompassed in the Composite Source Model by a repetitive cell pattern associated with each state. See Section 2.4.2.

2.4 The composite source model

As indicated in the section above, the activities of a source type are modelled by state diagrams and generation patterns. A more thorough presentation and discussion of the model is found in [21]. The properties of this model is studied in [22].

In the first subsection the source types are modelled. A source type may be regarded as the application of a specific service, e.g. voice telephony or file transfer. In Section 2.4.4 it is shown how a number of individual and independent sources with a behaviour defined by each of the source types may be included in a composite source model.

Before proceeding to the description of the composite source model, note that the above discussion may be summarised by the requirements for the model listed in the box *High level requirements for an ATM traffic load generator*.

2.4.1 State models

The activities on the dialogue- and burst levels are mostly driven by human activity, events in the environment of the system, composition of a number of events influencing the communication, etc. This forms a stochastic process. Hence, the dialogue and burst levels of a source type are modelled by a finite state machine (state model). Its behaviour is that of a finite state continuous time Markov process [23]. In its simplest form, each activity is represented by a single state, giving a negative exponentially distributed sojourn time of this activity. However, the sojourn time of an activity cannot always be properly modelled by a negative exponential distribution. Furthermore, it cannot always be modelled as independent of the previous state/activity. To handle non-negative exponential distributions, such activities are modelled by groups of states, which may have Coxian, Erlang, hyper-exponential or general phase type distributed sojourn times [2]. Dependencies may be introduced by letting the entry probabilities of a group be dependent on the previous state. See Figure 2.9 for an example. In the figure the "large task" has a non-exponential sojourn time when it starts after the source has been idle. Entered from above, it has a negative exponential sojourn time distribution.

Formally, the activity on the burst and dialogue level of source type k is modelled by a Markov process with transition matrix $\Omega^{(k)}$. Since it is more descriptive we have used the transition probabilities $p_{ij}^{(k)}$ and the state sojourn times $T_i^{(k)}$ in the model definitions, i.e.

$$\Omega^{(k)} = \left\{\omega_{ij}^{(k)}\right\}$$
$$\omega_{ij}^{(k)} = \frac{p_{ij}^{(k)}}{T_i^{(k)}} \qquad (2)$$

2.4.2 Patterns

Cell generation patterns model the cell stream resulting from the information transfer associated with each state. This stream is formed by the terminal and/or end system. Note that the parts of the system performing these tasks are rather deterministic "machinery". Hence, deterministic models of this activity suffice.

Each state has a pattern $\underline{\lambda}$ associated with it, see Figure 2.9. The pattern is a binary vector where 1 corresponds to the generation of a cell and 0 a cell length gap. As a trade off between modelling power and implementation we let this pattern be of constant length L and cells are generated cyclically according to this pattern as long as the source is in the corresponding state. Figure 2.13 gives an illustration of two patterns.

Constant length patterns give a reasonable model of the behaviour of the terminal and end system since this usually has

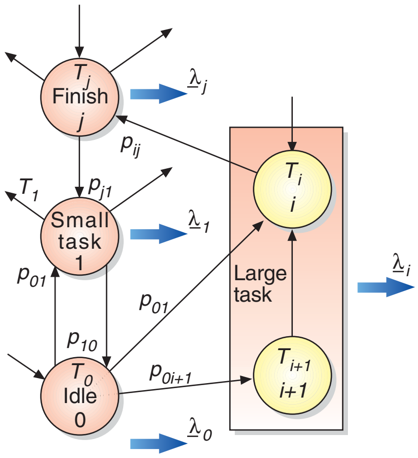


Figure 2.9 Generalized example of a source type model

a repetitive behaviour. In the generator we have two kinds of patterns:

- ordinary, with a length $1024 \leq L_B \leq 4096$ cells with a default value 2048. A pattern of the default length with one cell roughly corresponds to a payload transfer rate of 64 kbit/s on a 155 Mbit/s link. Most source types and states may be modelled with ordinary patterns
- long patterns, with a length $n \cdot L_B$ n = 2, 3, ..., 32. These are used to model longer cycles, for instance video frames.

2.4.3 A simple source type model example

To illustrate the model elements introduced above, a simple example is regarded. In Figure 2.11 a human user interacts with a high performance computer via an ATM network. For every user action, the computer transfers in average 40.8 kilobytes. For simplicity assume that the number of bytes in each transfer is negative exponentially distributed. The link rate is 150 Mbit/s and the effective rate out of the computer during a burst is

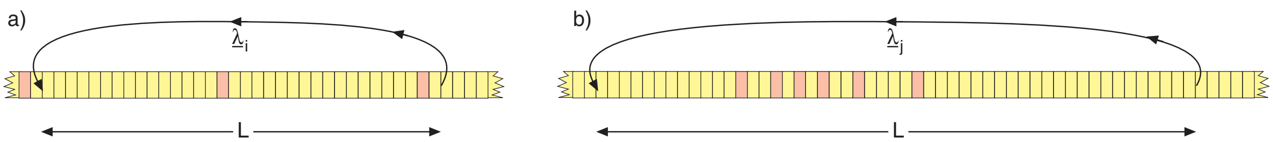
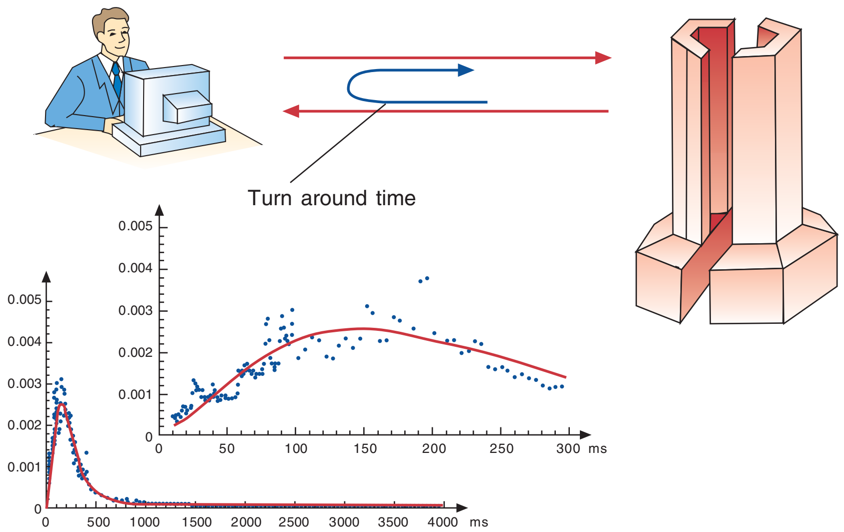
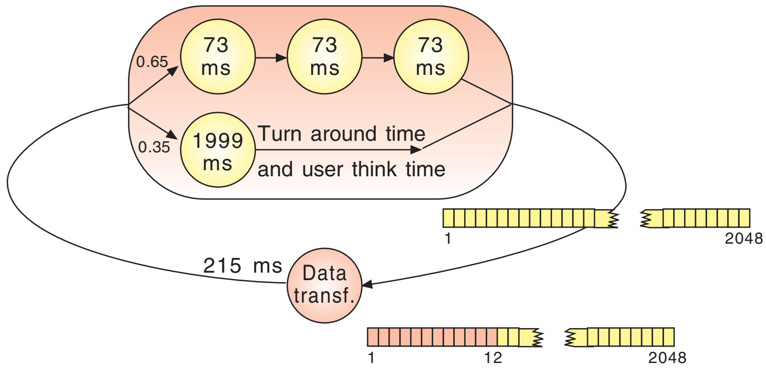


Figure 2.10 Examples of cell generation patterns
a) Equidistant cell interarrivals
b) Periodic bursts



a) ATM network user working toward computer with turn around times



b) Corresponding source type model

Figure 2.11 Example of a source type model

Table 2.1 Parameters governing the next transition of the composite model

The time until next state change is negative exponentially distributed with parameter	$\sum_k \sum_i \frac{m_i^{(k)}(t)}{T_i^{(k)}}$	(3)
The next type *k* and state *i* combination to be affected by a state change is given by the probability	$\frac{m_i^{(k)}(t)/T_i^{(k)}}{\sum_k \sum_i m_i^{(k)}(t)/T_i^{(k)}}$	(4)
The specific source among these which is changing state is given by the probability	$\frac{1}{m_i^{(k)}(t)}$	(5)
The next state *j* of this source is given by the probability	$p_{ij}^{(k)}$	(6)

1.52 Mbit/s. The data is transferred as 512 byte blocks, split into 12 cells, which are sent back to back.

The think and turn around time of the user is measured with results as shown in the figure. Two modes of the user may be identified, typing and thinking, represented by the upper and lower branch in Figure 2.11.b respectively. The number of states and the parameters of the user-submodel may be determined from the measured data for instance by a non-linear fit.

2.4.4 Composition

Up till now we have dealt with the modelling of a single source type. To meet the requirements for a useful generator, it is necessary to generate traffic from a number of sources which may be of different types. In the rest of this subsection, it is outlined how this is done.

Say we have K different source types with $m^{(1)}$, ..., $m^{(k)}$, ..., $m^{(K)}$ sources of the different types. Of the k'th type, let $m_i^{(k)}$ denote the number of sources in state i at time t. Since the composition of Markov processes also is a Markov process, the set of $\{m_i^{(k)}(t)\}_{\forall(k,i)}$ constitutes the global state of the dialogue and burst level part of the model.

The transition matrix for the composition of sources can be obtained by Kroenecker addition of the transition matrixes of the $m^{(1)}$, ..., $m^{(k)}$, ..., $m^{(K)}$ sources. The matrix become extremely large. A simple example for illustration is shown in Figure 2.12. The state space of three two state speech sources and one three state date source yields a 24 state global state space. The largest burst level state space which may be defined for the STG has 10^{4315} states.

However, by the principle used for generation, it is not necessary to expand this state space. The state space is traversed according to a set of rules given by the source type definitions and the number of sources of the different types. The selection of the next global event, state change and time when it occurs, is governed by the probabilities shown in Table 2.1.

The discussion of the state space above relates only to the burst level state space. In addition, a burst level state may be entered at an arbitrary time relative to the cyclic pattern described in Section 2.4.2. Hence, we must add these patterns together with a correct phase in order to produce the multiplexed output from the composition of the $m^{(1)}$, ..., $m^{(k)}$, ..., $m^{(K)}$

sources. The number of such multiplexes is obviously much larger than the number of burst level states.

The composite cell pattern transmitted from a multiplex of sources is denoted the periodic cell transmission pattern (PCTP). It is, with two modifications, the sum of the patterns of all the sources added together. When a source is added the identity (e.g. VCI) of the source is included. The modifications are:

- The start of the pattern is aligned with the position in the PCTP which determines the cell generated when the source entered its current state.
- If a cell from a pattern is attempted added into an already occupied position in the PCTP, it is shifted ahead to the first vacant position in the PCTP.

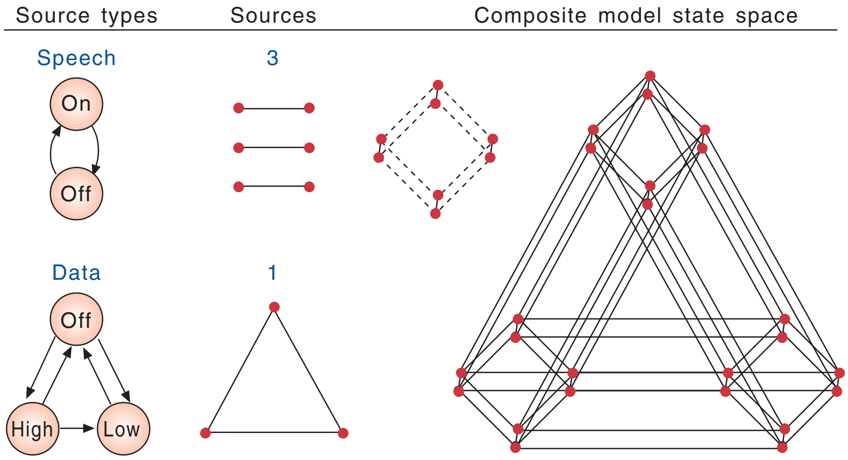


Figure 2.12 Illustration of expansion of the global burst level state space for a very small scenario

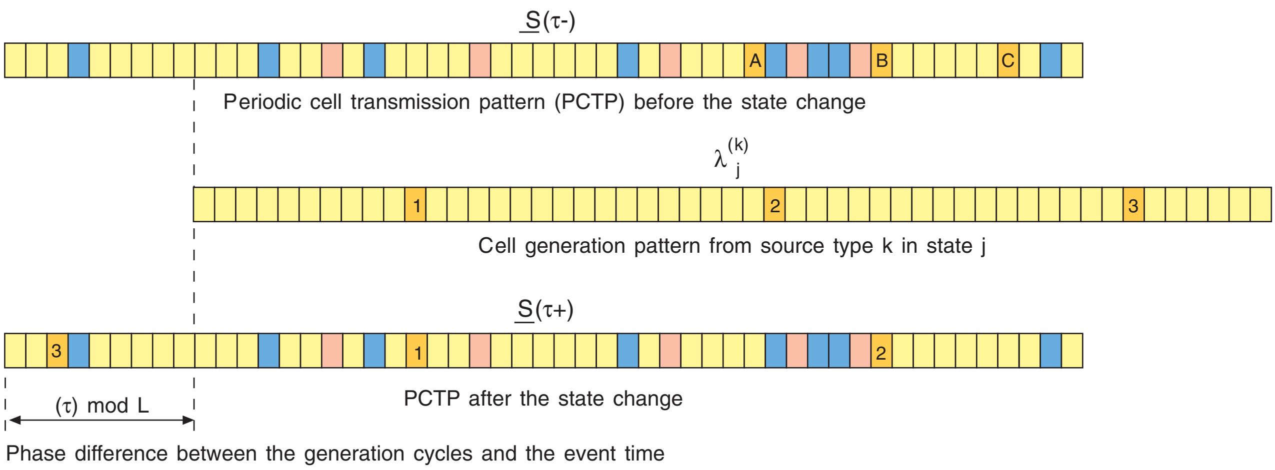
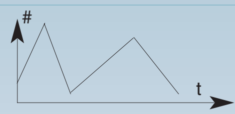
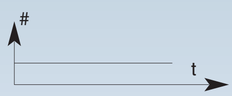
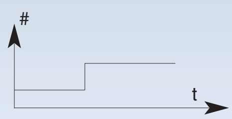


Figure 2.13 Example of the periodic cell transmission pattern (PCTP) immediately before and after a source changes state. Cells from the actual source are denoted A, B, C, before and 1, 2, 3 after

Table 2.2 Example of a scenario set-up table

Source Id	Fore-ground	Source type	Destin. tag	Cell header	Cell payload	Load Profile	
001–100		Phone	A	'Range of Names'	'Name'	Jumpy	
200	Y	HDTV	A	'Name'	'Name'	Const.	
300–350		File transfer	B	'Name'	'Range of Names'	Step	

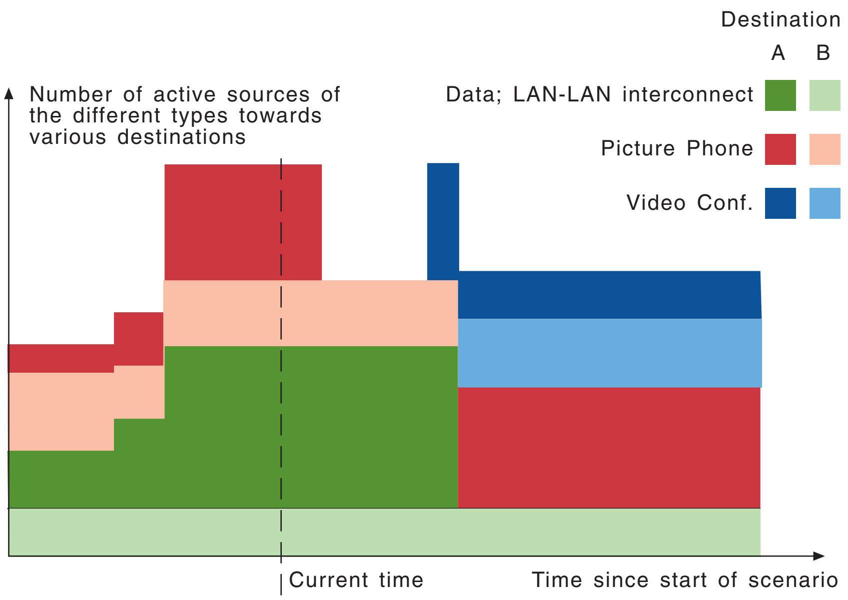


Figure 2.14 Example of scenario load profiles in terms of number of sources

This procedure is illustrated in Figure 2.13. For a formal description is referred to [21], where also a comparison of this multiplex scheme and a FIFO multiplex is given. If the PCTP at time t is regarded as a vector $\underline{S}(t)$, cells are transmitted cyclically from this pattern, i.e. at time t a cell from source $S_{t\mathrm{mod}L}(t)$ is transmitted. Under a typical load condition, the PCTP $(\underline{S}(t))$ will change several times during a cycle of length L.

2.5 Scenarios

The sources of the various types sending traffic toward the destinations in the network are defined in a scenario. A set-up is exemplified in Table 2.2. Most of the entries are self-explanatory. Foreground sources are sources on which it is performed end-to-end measurements by the ATM 100 system, presented in Section 4.1.1. The destination tag is an indicator derived from the VCI/VPI in the cell header which highlights the terminal – or a point within the target system, the traffic is routed to. The load profiles need a further explanation. The load profiles model the connection levels as indicated in Figure 2.8. During generation they define the relative number of sources which are active at the various times. By active is meant that they are generating cells. See Figure 2.14 for an example on how six profiles govern the traffic load during a measurement.

In Figure 2.15 all the elements introduced above which constitute a scenario are illustrated. The "other parameters" are measurement period, random number seeds, initialisation, which generator to run it on, etc.

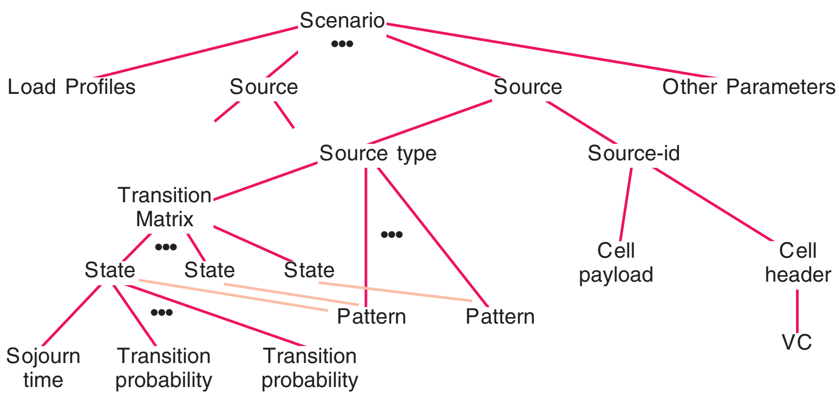


Figure 2.15 Elements contained in the traffic model constituting a scenario

3 Defining source type models

The identification of the various activity levels was discussed in Section 2.3. When a specific source is going to be modelled, it is usually obvious which activities should be modelled at the various levels. However, there may be cases where a type of activity may for instance either be modelled on the cell level or on the burst level. This section will give some guidelines and discuss more specific approaches to establishing source models. Since the dialogue and burst level are modelled by the same mechanism, no separation is made between them. Using Figure 2.7 as a basis we add some detail to obtain a rough outline of which elements of the source should be modelled by state machines and which should be modelled by patterns. An example is shown in Figure 3.1. A summary box for this section is found below, headed "How to use the Composite Source Model to describe source types". Source type modelling along these lines is also discussed in [24].

3.1 Burst and dialogue level modelling

The most important elements constituting the burst and dialogue level activity are

- user behaviour
- slow network interactions, and
- variation in the information source (for instance variable bitrate coded video).

When finite state machines are used as means for modelling, sources where the activities can be represented with distinct system states are more straightforward modelled than sources where (part of) the activity is continuously varying. These two cases are discussed in the separate subsections below.

3.1.1 Distinct system states

At any time the source, or end-system, performs one out of a finite number of different activities. These should be identified and characterised by *system states*. Thereafter, the duration of these states should be obtained as well as the transition frequency between them. It should be checked if there are any dependencies between the various transitions, between the various sojourn times, and between transitions and sojourn times. If such dependencies are found and considered important, they should be accounted for by splitting system states into more generator states, using phase type distributions etc. as indicated in Section 2.4.1 above. Similarly, non-negative sojourn times of system states should be modelled by phase type distributions. This is most easily illustrated by examples.

Example; Speech with silence detection

During a speech conversation, one party is either speaking or listening. The speaking party can either form sound or be pausing. In addition, we may have occurrences of interruptions and double talk. Furthermore, the behaviour of the parties may be dependent on who spoke last, etc. This is studied thoroughly in the classical paper of Paul T. Brady [25]. His six state model is reproduced in Figure 3.2.a. Brady found that the sojourn time of the states could (with a reasonable accuracy) be approximated by a negative exponential distribution.

Regard a speech encoding scheme where the sending of samples stops during pauses. In this case, we may regard the source as having two main states (Talking and Silent) where each of them may be split into three sub-states as shown in Figure 3.2.a. If these sub-states have negative exponentially distributed sojourn times, they may be modelled as phases. Brady found that the unconditional talking and silent periods were roughly negative exponentially distributed. This leads us to a simpler model shown in Figure 3.2.b. This simpler model is often used for mathematical analysis. See for instance [26] and [27]. Note that by going from the model of Figure 3.2.a to the simpler of Figure 3.2.b we lose model accuracy with respect to

- the sojourn time distribution of talking and silent periods, and
- dependencies between the two kinds of activities caused by interruptions etc.

Example; Interactive data transfer

This example is already presented as the simple source type in Section 2.4.3. Figure 2.11 shows how the duration of the system state "Turn around and user think time" is modelled by several states. Due to lack of a more detailed information from the measurements, it is assumed

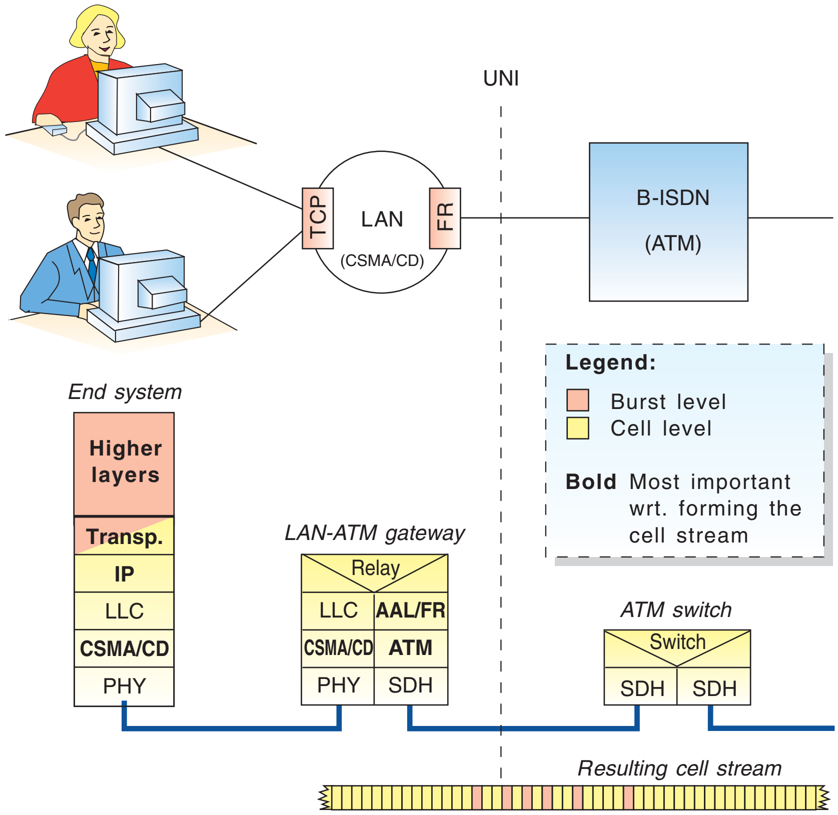


Figure 3.1 Rough indication of which parts of an end system that contributes to the various model levels

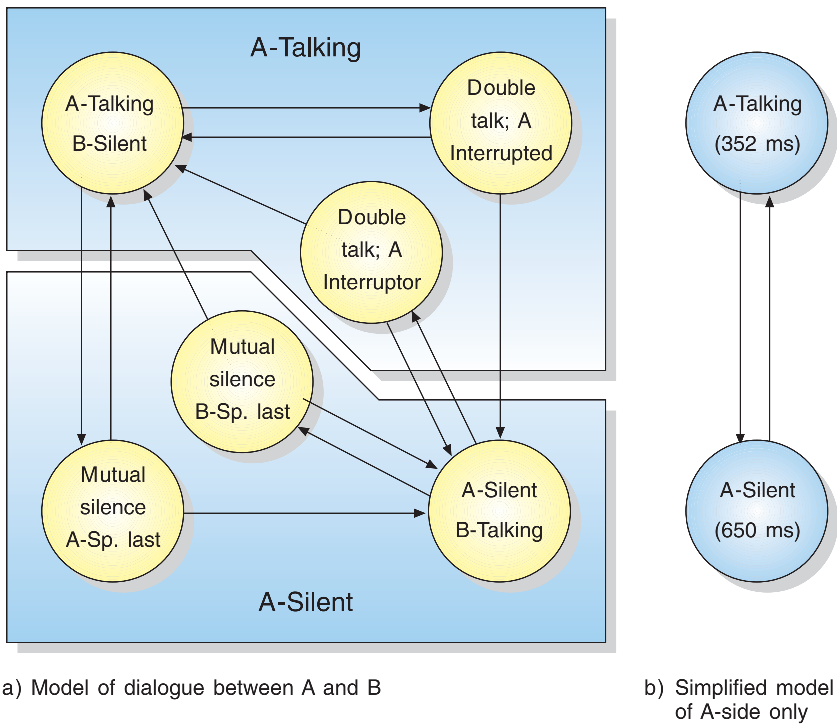


Figure 3.2 Brady's speech source-type model

that these times are identically and independently distributed. If this later turns out not be the case, the model has to be refined.

Example; Bulk data transfer towards a slow receiver

In the last example we regard a bulk data transfer, for instance an FTP file transfer, see Figure 3.3.a for an illustration. To demonstrate some of the modelling concepts, it is assumed that a window based flow control is used. In a bulk data transfer, for instance with FTP, the receiving computer, workstation or the local network on the receiving side may not be capable of consuming the data at sender speed. Hence, the window becomes full and sending is deferred. The sender must wait for acknowledgement or a wider window.

Measurements of bulk data transfers show that there is an extremely large variability in the size of the data items transferred [28]. This leads us to the system state model shown in Figure 3.3.b. In the system state *In-between items,* the activity is under user control. To keep the example simple, this part of the model has not been detailed. When a transfer is started, a *Send first segment/short file* system state is entered. This may be left either because it was a small data item which is transferred without filling the window, or because the window becomes full. In the latter case, the transfer has to be halted while the sender waits for acknowledgement, or, alternatively, a wider window is negotiated. The mean sojourn time in this system state is given by the turnaround time for that connection, which may be predicted given a specific installation and implementation. After reception of acknowledgement the sender resumes the transfer. The *Send rest of item* system state models the process after acknowledgement is received. The source stays in this state until it has to wait for acknowledgement again or the transfer is finished. This state is separated from the *Send first segment* state since it has a different sojourn time.

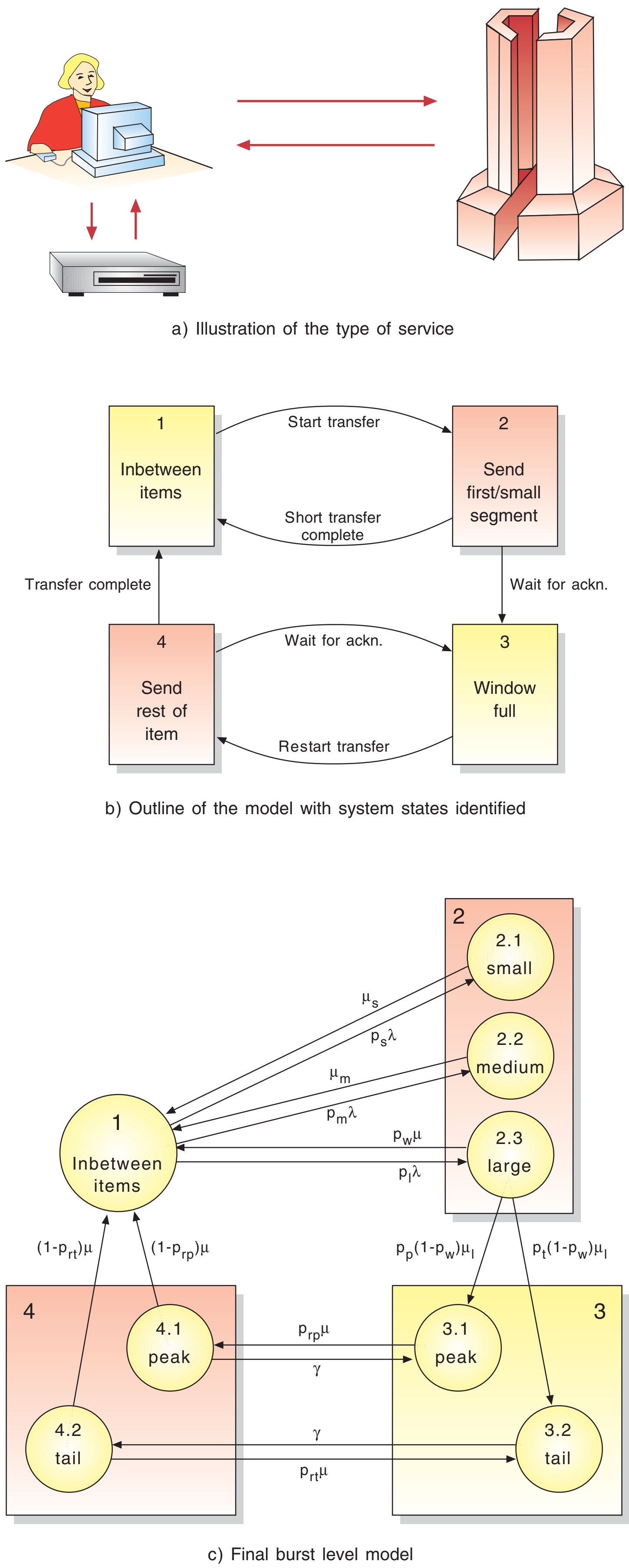
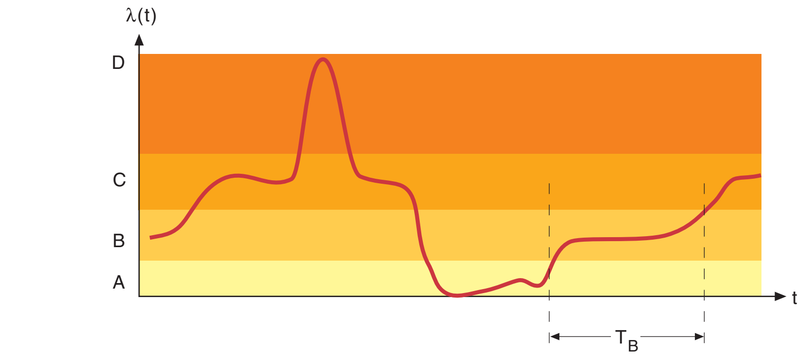


Figure 3.3 Illustration of a burst level model for bulk data transfer with a receiver which is slower than the sender
a) Illustration of the type of service
b) Outline of the model with system states identified
c) Final burst level model

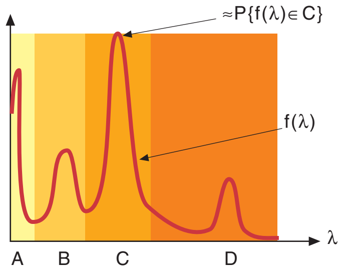
Figure 3.3.b shows a model outline for a slow receiver. For a <u>slow sender</u> the transfer will never have to stop and wait, and the transfer may be modelled by the first two system states.

Figure 3.3.c shows how the system states are detailed to fit the item size distribution by a phase type distribution. Empirical data from [28] shows that FTP-item distribution does not follow a negative exponential distribution. It is therefore necessary to refine the *Send* … system states to fit the empirical data. A closer examination of the distribution revealed that approximating it with a five branch hyperexponential distribution gave an acceptable fit, however, with deviations from the empirical data for the protocol specific "outliers" like items 512 bytes long.

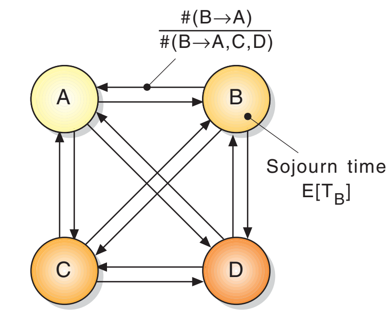
The transfer of items "belonging" to the two branches of the hyperexponential distribution representing the largest items are chopped into smaller segments by the *Window full* state. For the simplicity of the example it is assumed that an acknowledgement is received after a negative exponentially distributed time with mean λ^{-1} and that the transfer time of a segment is negative exponentially distributed with mean μ^{-1}. These distributions may easily be replaced by more realistic Erlang distributions. Note also that the "chopping" due to waits for acknowledgement, the two branches must be kept separate.



a) The information rate from the source with identification of information rate levels



b) The probability density (pdf) $f(\lambda)$ of the source having bitrate λ at a random instant and the approximation level probabilities.



c) The state diagram derived from source behaviour. State corresponds to rate level

Figure 3.4 Definition of a state diagram from source types without distinct states

3.1.2 Continuously varying activity level

Some source types do not have distinct system states. An example of such a source type is variable bitrate coded video. The procedure below indicates how such a source may be approximated by a state model by making the process discrete. The procedure processes an available sample output from the source. A video source model with good performance, defined according to a procedure similar to the one below, is reported in [29].

Smoothing

First, regard the information/bit-stream from the source as a continuous process. If it is not already smooth, the process should be smoothed, for instance over a moving window of a duration similar to the typical time constant reflecting the burst level, e.g. 20 – 200 ms. Such a smooth stream is depicted in Figure 3.4.a.

Level identification

The next step is to identify any levels in the information rate from the source. Such levels are recognised in Figure 3.4.a. An aid in identifying these levels is to plot the frequency of the various information rates. This is illustrated in Figure 3.4.b, where recognised levels are seen as peaks.

In cases where the source does not exhibit levels and corresponding frequency peaks, the range of the rate should be split into appropriate levels. The higher the number of levels, the higher the potential modelling accuracy.

Each information generation level corresponds to one state in the source type definition. This is illustrated in Figure 3.4.c. The detailed behaviour of the source when it is in a certain state (within a level) is encompassed in the pattern of this state as discussed in the next section.

Intralevel transition statistics

The relative numbers of jumps between the various levels correspond to the state transition probabilities. When a level is only passed, i.e. the duration of the level is significantly shorter than the average, the level should be omitted. For instance in Figure 3.4.a we take the following jumps/transitions into account: $B \rightarrow C \rightarrow D \rightarrow C \rightarrow A \rightarrow B \rightarrow C$, and the estimate of the transition probability from B to A based on this small sample becomes

$$p_{BA} = \frac{\#(B \rightarrow A)}{\#(B \rightarrow A, C, D)} = 0$$

Correspondingly, the transition probability from B to C is one. A real estimation requires a far larger number of transitions.

Level duration

The state sojourn time corresponds to the time the source stays within one level. See Figure 3.4.a for an example of one observation of a sojourn time, T_B. Hence, from the sample output of the source, the distribution of the state sojourn times may be estimated, i.e.

$$P\{T_B \leq t\} = \frac{\#(T_B \leq t)}{\#(T_B)}$$

If the sojourn time distribution is sufficiently close to a negative exponential distribution, i.e.

$$P\{T_B \leq t\} \approx 1 - e^{-\frac{t}{E(T_B)}}$$

the level may be modelled by a single phase state. What is accepted as "sufficiently close", depends on modelling accuracy and the constraints put by the STG hardware, for instance the number states available in a scenario discussed in Section 4.2.2. If the sojourn time distribution differs significantly from a negative exponential distribution, we may split the state into a number of phases similar to the examples in Section 3.1.1 above.

Following this procedure, a Markov model representing the burst level of the source without distinct states is established.

It should also be checked for dependencies between the sojourn times on the various levels and the next level/state. If, for instance, there is a positive correlation between long sojourn times in state *C* and transitions into state *D* (and between shorter sojourn times and transitions into *B* and *A*), we may include this into the phase type model. See Figure 3.5 for an example. In this case, transitions to *A* and *B* will take place after the sojourn time of the first phase (negative exponentially distributed) and transition into *D* after the sojourn times of both phases (generalised Erlang-two distributed). Other combinations of sojourn times – transition dependencies may of course also be constructed.

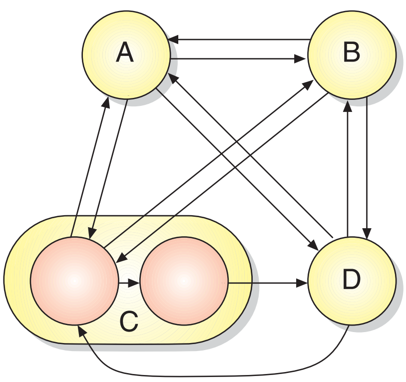


Figure 3.5 State C is split into two phases to include sojourn time transition dependencies

Validation

A check of whether reasonable sojourn times and transition probabilities are obtained or not, may be obtained by

- computing the steady state probabilities
- dividing them by the width of the respective levels, cf. Figure 3.4.b, and
- comparing the result to the level frequency plots.

In the above procedure, it is implicitly assumed that the level/state changing procedure of the source type has a (semi-) Markov property. This may be validated by a direct testing of the properties of the embedded discrete time chain [30].

The long term dependencies may also be validated by a comparison between the autocorrelation of samples from a source of this type, and the autocorrelation of the source type model. (Note that the cell level should be smoothed.) See [22] for a computational procedure.

3.2 Cell level modelling

The most important factors determining the cell level behaviour of a source are:

- the processing, data retrieval and transfer speed of the end user equipment
- the lower layer protocols, their mechanisms and implementation
- contention in user equipment and customer premises network, e.g. a CSMA/CD based LAN, and
- possible traffic forming mechanisms, see [31] for a discussion.

Based on the experiences from the cell level modelling so far, a framework for building cell patterns will be outlined in this section. Focus will be on the major influences from the various protocols in the protocol hierarchy as well as the other facts mentioned above. For some source types, not utilising all protocol levels only parts of the discussion apply. An example is variable bitrate video transfer presented directly to the AAL. However, the basic approach should apply. The parameters may be obtained by deduction from system and protocol descriptions, from measurements of systems and protocols if these are available, or by a combination of these approaches.

A cell pattern consists of a set of *minibursts,* or *subpatterns.* These are usually, but not necessarily, regular. Minibursts arise from higher level protocol data units which propagate down through the protocol levels and result in a short burst of cells. Figure 3.6 shows a generic regular cell pattern with its descriptive parameters, a 4-tuple $\{I_C, I_B, L_B, L_P\}$. In the following we focus on the different parameters in the tuple, except for the length of the pattern, L_P, which is chosen to fit the regularity described by the other parameters as well as the design of the STG, cf, Section 4.2.3.

3.2.1 Peak cell rate

The peak cell rate is the inverse of the minimum inter-cell distance, i.e. $1/\mathrm{Min}(I_C)$. The inter-cell distance or cell spacing depends primarily on the implementation of AAL (ATM Adaption Layer). Two extremes may be identified:

- *Maximum stretching,* as described in [31]. This implies that cells are spaced as evenly as possible. The allowable spacing will depend on the real time requirement of the source/user. Usually, these are so slack that cells may be evenly spaced over an entire pattern,

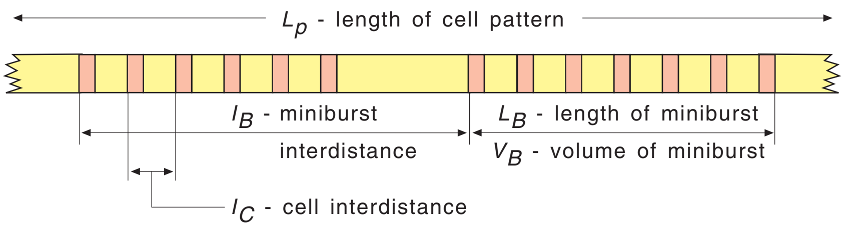


Figure 3.6 Parameterized framework for building cell level models based on patterns

which typically lasts for 5 ms. This, however, requires a co-operative flow control between various protocol layers.

- *No spacing*. When a PDU (Protocol Data Unit) arrives at the AAL, the entire PDU is sent continuously, i.e. cells are transmitted back-to-back. This will cause compact peaks (minibursts) in the cell pattern, shown in Figure 3.6.

The latter obviously results in a much worse traffic for the network to handle. During the establishment of a connection a maximum allowable peak rate is negotiated between the network and the user. This will usually correspond to something in-between the two extremes above. This negotiated rate r_p will determine the intercell distance

$$I_C = r_P^{-1}$$

Note that this must be the case for all the patterns related to a source, irrespective of the corresponding average load of the pattern, as long as there is no mechanism to inform the AAL whether the source is in a "peak state" or not.

Depending on the "regularity" of the equipment there may be slight variations of the value of I_C during a miniburst and pattern.

3.2.2 Miniburst volume

The number of filled cells in a miniburst is obtained by considering the following items:

- The minimum Protocol Data Unit (PDU) size the traffic is split into, V_{PDU}. For instance, keeping Figure 3.1 in mind, the FTP traffic over TCP/IP, CSMA/CD (Ethernet) and Frame Relay have a minimum PDU size of 1500 information bytes + 26 header bytes given by the CSMA/CD frames.
- The segmentation and reassembling strategy at each level in the hierarchy. For instance, are small PDUs reassembled to larger PDUs at any level? Still keeping Figure 3.1 in mind, Frame Relay has a maximum segmentation size of 1600 bytes. Whether two packets (PDUs) of e.g. 600 bytes are reassembled to one Frame Relay PDU or not, is a question of implementation and may therefore not be generally stated.

The V_B may vary within a cell pattern.

How to use the Composite Source Model to describe source types

- Fitting to measured user behaviour *Section 3.1.1 and Section 2.4.3*	State models
- Varying information transfer rate or activity level · distinct states *Section 3.1.1* · continuous *Section 3.1.2*	State models
- Modelling measured data transfer volumes and anticipated end-to-end protocol behaviour *Section 3.1.1,* third example	State models
- Packet formation, segmentation and reassembly · protocol stack · intermediate LAN *Section 3.2.2*	Patterns
- End-system and CPN performance and contention *Section 3.2.3*	Patterns
- AAL protocol behaviour – cell spacing *Section 3.2.1*	Patterns

3.2.3 Minibursts interdistance

The minibursts interdistance, I_B, i.e. the number of cell periods or duration between start of minibursts depends on a number of factors, as well. While the minibursts volume is primarily dependent on the behaviour of the protocols involved, the miniburst interdistance is primarily dependent on the rate of information transfer and the speed of the protocol execution.

- The rate of information which shall be transferred. If, in a certain state, a_i bit per second is going to be transferred, it is seen that this must correspond to the ratio between miniburst volume and inter-miniburst distance for the pattern of this state, i.e.

$$a_i = \frac{E(V_B)}{E(I_B)}$$

- The service rates of levels in the protocol hierarchy which the PDU passes on its way from the application to the end level, may influence the inter-miniburst distance. The service rate will be determined by equipment speed, protocol implementation, transmission speed in a CPN (Customer Premises Network) and similar. The minimum/ slowest of these will be the dominate factor in determining I_B. Note that this factor may determine the information transfer rate, a_i, as well. For instance, in a configuration like the one in Figure 3.1, the raw transmission rate of an Ethernet is 10 Mbit/s which may be the limiting factor for some services.
- In addition, contention in a CPN, e.g. collisions in a CSMA/CD based LAN, may cause variations in the inter-miniburst distance and reduce the maximum effective service rate. For instance, the back-off retransmission strategy in Ethernet may introduce variation between Protocol Data Units. In this context, however, this is significant only for a heavily loaded network with a high probability of collision. In general, contention in a CPN may be modelled by a variable inter-miniburst distance I_B within a cell pattern where the degree of contention is reflected in Var(I_B) and Max(I_B) - Min(I_B).

4 The synthesised traffic generator

In the above sections, the more theoretical and principal aspects of generation of load for ATM measurement have been dealt with. In this section a generator system based on this principle will be presented. In Section 4.1 its design is outlined and in Section 4.2 the capabilities

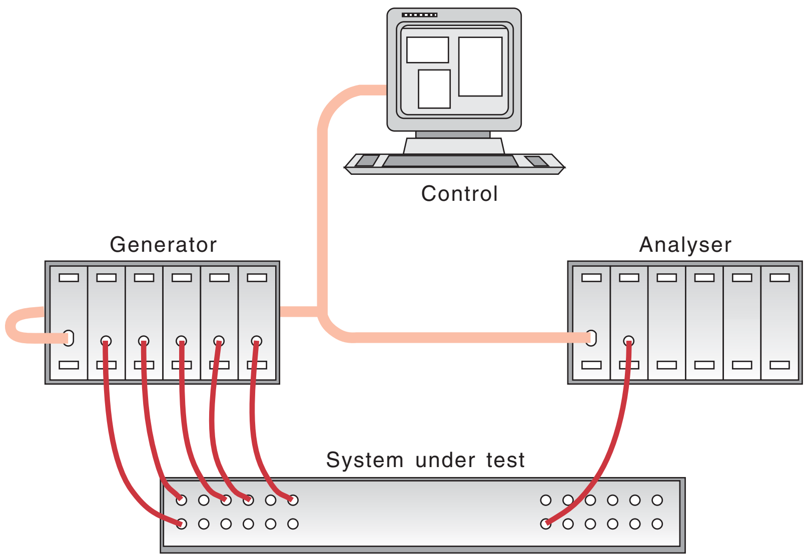


Figure 4.1 The ATM 100 measurement system with multiple STGs in a bulk load configuration

are summarised, before potential extended functionality is presented in Section 4.3.

4.1 Design

4.1.1 The measurement system

The work on the STG was started during the RACE project PARASOL (R 1083). PARASOL was established to define measurement and validation methods for ATM systems and to develop prototype equipment. At the end of the project, beginning of 1993, a complete modular measurement system was available. For a more detailed presentation of PARASOL see [32] and [33].

After the conclusion of the PARASOL project, its prime contractor Wandel und Golterman (D) started work on turning the prototype into an industrial pilot series named ATM 100. This effort included a complete redesign and re-development of the measurement control system. Telenor Research and DELAB have carried out this work related to the STG as a part of an agreement between NTR and W&G. This enhanced system is now running in a number of sites.

An overview of this system is given in Figure 4.1. It consists of three parts. The *generator* which produces load, including test cells, towards the system under test (SUT), the *analyser* which analyses the test cells from the SUT and the control workstation which is used for setting up the instrument, control during measurements and for presentation of the results. These three parts may reside in the same physical cabinet, or they can be distributed over a wide area, e.g. to measure the traffic flow through a network. In addition, the X-window based MMI allows remote operation of the control system.

The measurement system is based on modules with different functionality, as indicated in the figure. Each module performs a specific task. For instance, important modules of the *analyser* are

- the module which traces a subset of the connections or a complete cell-stream, and
- the module which performs a real time analysis of properties of the cell stream and presents them on the MMI.

This modularity makes ATM 100 a flexible tool, which is easily reconfigured to perform a wide range of measurements, like bulk load generation as in Figure 4.1, and real time analysis of live traffic[2]. In the following it will be concentrated on the STG.

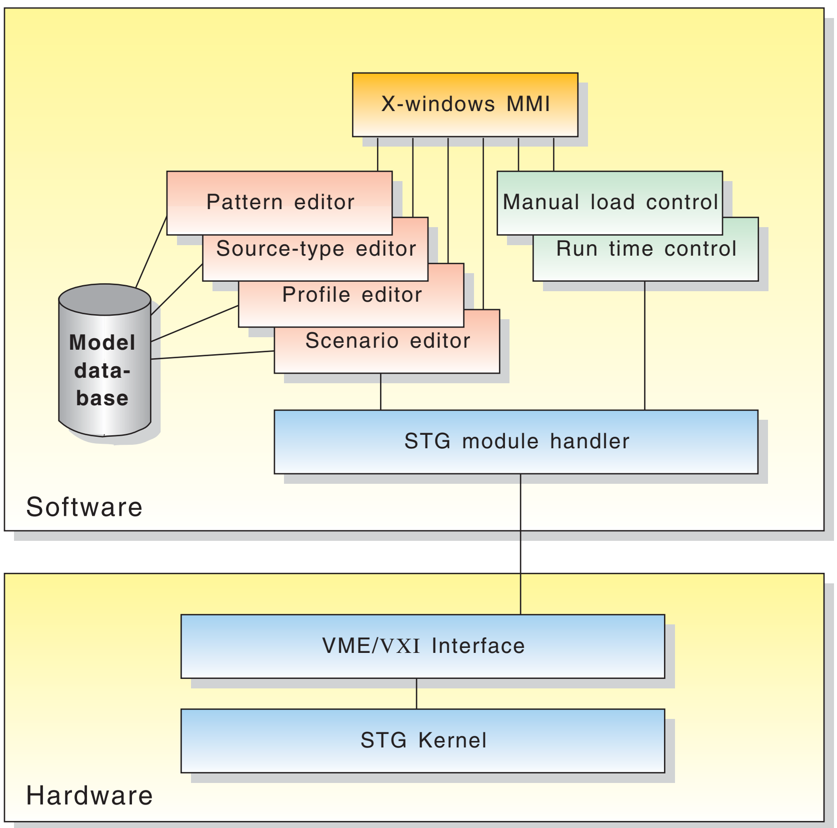


Figure 4.2 STG system block diagram

[2] *For further information about the ATM 100 measurement system, we refer to Wandel und Golterman.*

4.1.2 The STG overview

The purpose of the STG is to generate traffic with properties similar to a multiplex of independent sources where each source has a behaviour close to real source. The software and hardware elements that constitute the STG sub-system are shown in Figure 4.2.

The software runs on a SUN SPARC under SOLARIS 2.3. It includes an Open Windows based man machine interface (MMI), several graphical editors, the STG Module Handler performing the necessary checking and compilation of source types and scenarios and the STG specific drivers. The system also contains the STG database which includes various set-ups, scenarios, source-type definitions, etc.

4.1.3 The STG hardware

The hardware is made as a two slot C-size VXI module. It consists of 12 FPGAs (field programmable gate arrays), each in the range 10,000–20,000 gates and 10 Mbit of RAM. One of the slots serves as the VME/VXI interface, which handles the communication with the rest of the instrument. The other slot is the STG kernel and handles the composite source model. The functionality is arranged as shown in Figure 4.3.

In the block to the left, the next time and old/new state are found for the next state change, cf. (3) and (4) of Table 2.1. These are transferred to the next block, and the block is ready for the next operation. The FIND SOURCE block finds the source for the state change, cf. (5) and (6), and finds the pattern associated with the new state. These data are sent down to the UPDATE PATTERN block together with the time when the pattern should be updated. The FIND SOURCE block also handles the activations/deactivations sent by the software, and the long patterns. Long patterns are handled as a set of normal patterns resulting in a "state change" for each new part of the long pattern.

The UPDATE PATTERN block updates up to 8 state changes in parallel, at the time requested. The old patterns are removed from the PCTP (periodic cell transmission pattern) and the new ones added, see Figure 2.13.

Every 2.7 microsecond a source_id is read out from the PCTP and sent to the ASSEMBLE CELL block. This block uses the source_id from the PCTP as a pointer to a cell with header and payload,

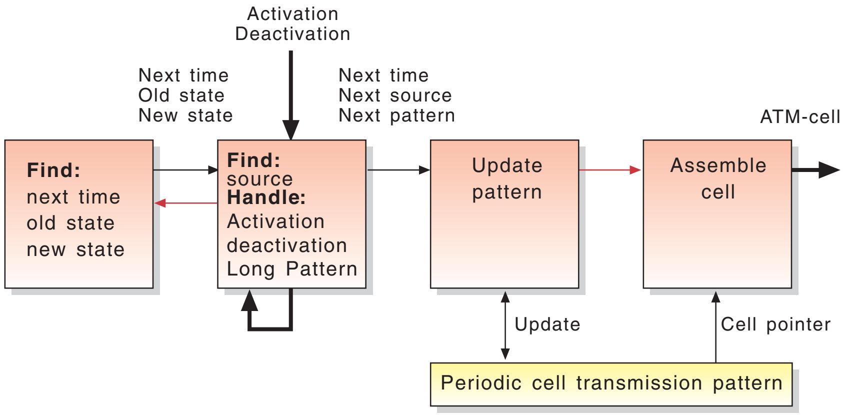


Figure 4.3 The STG hardware block diagram

and sends this cell out to the next module in the test tool. A cell may be transformed into a test cell (foreground traffic) before it ends up in the output module (OM).

The two blocks to the left in Figure 4.3, which handle the stochastic parts of the model, use four Tausworthe pseudo-random generators with primitive trinomials between 25 and 34 bit [34, 35]. The sequence lengths of each random generator are mutually prime. To ensure no correlation between the numbers, the generators are clocked as many time as their length between each time a number is used.

The activation and deactivation are used to emulate the connection level. The software sends activation and deactivation messages to the hardware at the time these should take place according to the load profiles. There will be one profile for each pair of source type and destination in the network, see Table 2.2 for an example. In a profile the number of sources for this pair will vary according to the graph. When the level changes, the software selects which sources to activate/deactivate. Since the properties of all the source of the same type toward the same destination are identical, an arbitrary selection of these may be chosen. The software handles simultaneously all profiles for all source types toward all destinations, cf. Figure 2.14. When a profile changes, the activation/deactivation which corresponds to the change will be sent to the hardware. The changes are effectuated in the hardware as soon as possible, which is maximum eight activations or deactivations per 0.1 ms and, on the average, seven at nominal load.

4.1.4 The STG software

The software is written in C. The volume of the executable is about 12.5 Mbytes and on-line help, database, etc. occupy a similar volume. An outline of the various software elements is presented in Figure 4.2.

The *Scenario editor* is in many ways the core of the STG software. For each measurement, a *scenario* is defined. In this editor the number of sources, source types, destinations, cell headers/payloads and load profiles are put together, see Table 2.2. It also checks the parameters and calculates values, like average and maximum load, average rate of state and pattern changes. Graphs like the expected load as a function of time and type/destination are also calculated and presented to the user, see Figure 2.14.

Load profiles for activation – deactivation, i.e. handling the connection level, are defined in the *Profile editor* as a graph of how many sources that are active at any measurement instant. The graphs are similar to those indicated in Table 2.2.

The hardware is initialised and controlled through the *STG module handler*. When a measurement shall be carried out it calculates the necessary parameters to set the STG in a steady state for the actual load scenario, transforms the scenario definition to the format required by the STG and downloads the data to the hardware.

The *Source type editor* allows an efficient definition of state diagrams with a "bubble" for each state containing name, sojourn time and a pattern_id, arrows between the states hold the next state probabilities, i.e. similar to what is indi-

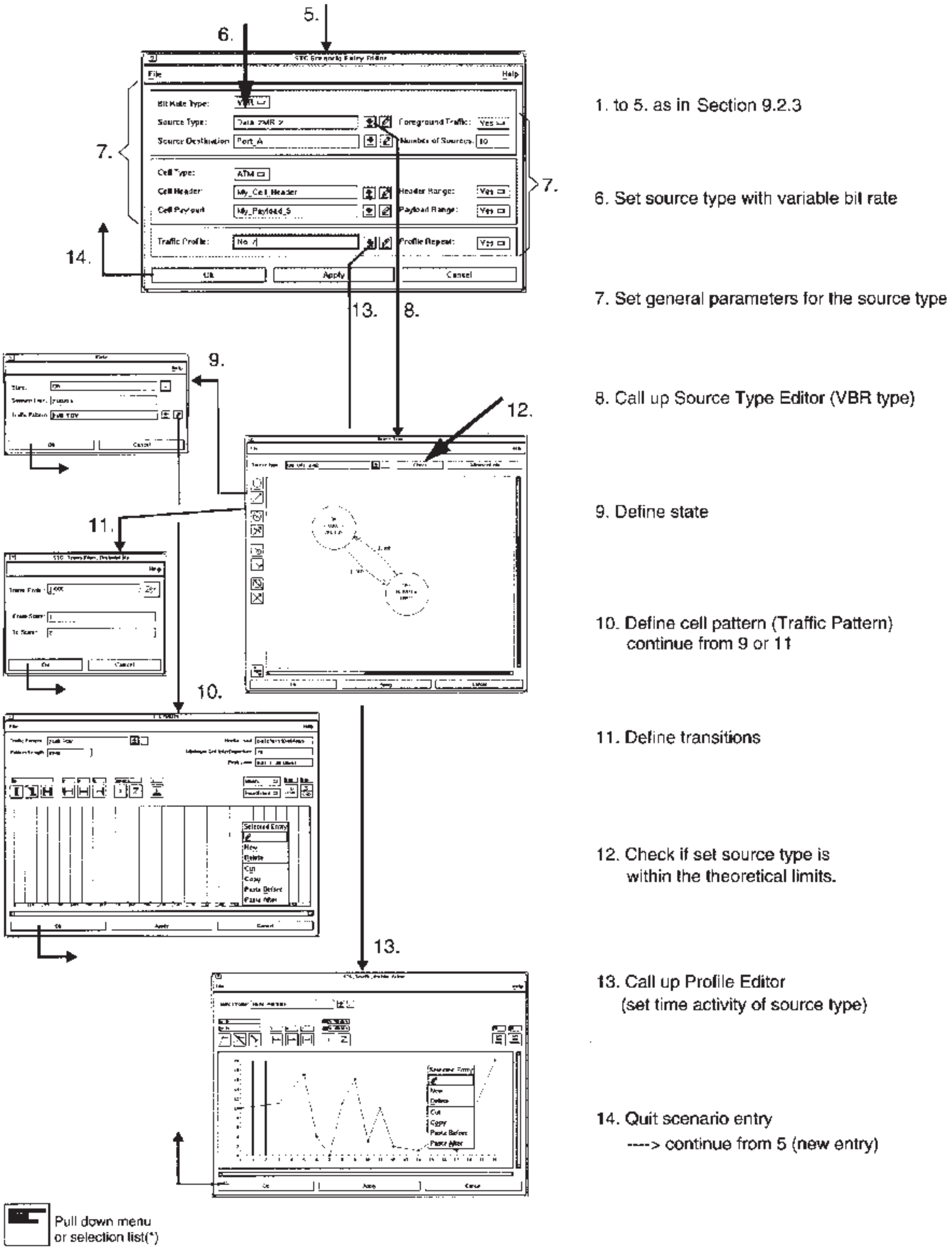


Figure 4.4 Excerpt from the ATM 100 on-line manual "Synthesized Traffic Generator Module; 9.2.5 Direct selection of the editors"

cated in Figure 2.9. After a check the diagram may also be toggled to show the load contribution of the states, phase type distributed pattern holding times, and, for model debugging purposes, whether the state is absorbing, transient, included in a disconnected chain, etc.

The *Pattern editor* defines graphically the pattern as a bitmap on the lowest level. The bitmap looks similar to the cell level in Figure 2.13. To define a pattern bit by bit is a demanding task. Hence, the editor has functionality for generation of patterns and subpattern automatically according to various intercell distributions, as well as copy, paste, paste repeatedly, etc. of subpatterns. It is possible to zoom in and out for the required accuracy and overview.

To give an impression of the man machine interface (MMI) of these editors, an excerpt from the ATM 100 online manual is given in Figure 4.4.

When a load scenario is executed, the *Run time control* performs activation – deactivation according to the defined load profiles. A graph similar to Figure 2.14 is presented on the user interface with a vertical bar indicating where the generator is in the profile.

The execution of a predefined profile may be interrupted at any time, and the *Manual load control* is switched on. This control enables the operator to activate or deactivate arbitrary sources, to increase or decrease the load from a specific source type or toward a specific destination or a combination of these by pushing "knobs" or by giving a numeric value of the desired load.

All of these functional modules use a common database containing definitions and libraries. For instance, if an interesting load mix is found during manual control, the mix may by entered the database as a scenario.

4.2 Generator capabilities

Although technology and cost constraints restrict the capabilities of the STG and design trade-offs have to be made, the current version of the STG is expected to cope with most of the user requirements. This section summarises the capabilities of the STG.

4.2.1 Number of different sources

One module is capable of generating traffic from up to 2047 independent sources. For comparison, if 64 kbit/s CBR sources occupy all the link capacity, a 100 % utilisation for this purpose would require 2175 sources. Taking into account that many ATM sources will require more than 64 kbit/s, it is believed that 2047 sources more than cover the need for describing the load on one link. For extreme requirement more sources can be provided by merging traffic from two or more adjacent STGs in a mainframe.

4.2.2 Source type complexity in a scenario

A source type characterises sources with identical statistical behaviour, see Section 2.4. The STG can handle 128 states, which may be used for ten source types, each comprising 12–13 states. This was regarded as sufficient. The user is not restricted in the utilisation of the available states, but may define anything between the extremes a single source type of 128 states or 128 types each having one state.

4.2.3 Pattern lengths and long patterns

The traffic stemming from one source in an active state is contained in a pattern, cf. Figure 2.10. As default the length of this pattern is 2048 cell slots. This corresponds to a minimum payload rate of 69 kbit/s. If one anticipates that a narrow band constant bitrate (CBR) service, like conventional PCM telephony, should constitute a significant part of the traffic in the ATM network it becomes important to generate this type exactly. This can be accomplished by setting the pattern to the length of 2193 cell slots. The pattern length may be varied between 1024 and 4096 cells.

Some source types may have specific cell transmission sequences exceeding the maximum pattern length. An example of such a source is a variable bitrate (VBR) video codec and transfer system exhibiting a frame periodic behaviour. The STG is provided with long patterns to allow the user to model such events. A long pattern is simply a concatenation of between 2 and 30 ordinary patterns thus covering a cell sequence lasting for up to 0.168 s[3].

[3] *It should also be mentioned that the STG may serve as a deterministic pattern generator with a period equal to that of the maximum long pattern. Thereby the STG also offers the features of debugging tools like memory based traffic generators.*

4.2.4 Sojourn times and activity rate

The ratio between maximum and minimum mean sojourn times of the states of source types in a specific scenario is limited by the number of digits in various variables in the hardware arithmetic of the STG. It will also depend on the number of sources specified by the user. A scale factor ranging from 0 to 15 allows the range of the sojourn time to be altered as shown in Table 4.1. *The time between events at burst and dialogue level has a granularity of* 2.7 µs *and can be up to 3.5 hours irrespective of the scenario size.* To avoid overload of the hardware by excessive source activity, the minimum mean state sojourn time is set to 100 µs.

4.2.5 Random number generation

For verification purposes, it is mandatory to load the system under test with non periodic and non repetitive traffic. The variates used in the STG, cf. Table 2.1, are generated by feedback shift registers [34]. The periods for these are 2^{33} - 1, 2^{34} - 1, 2^{25} - 1 and 2^{31} - 1, which are relative prime numbers. The number of shifts between draws are 33, 34, 25 and 31, respectively, and they are not divisible by the cycle length. Thus the number of state changes (variate sets) before the sets of numbers from the four generators will be repeated, equals the product of the individual generator periods. If the nominal value of one state change per 0.1 ms is observed, the STG will produce non repetitive traffic for 33×10^{24} years. The traffic will then be repeated only if all states contain exactly the same number of sources as when the STG was started, which is most unlikely.

By reading the content of the shift registers at the end of a measurement block, and using these as generator seeds for the next experiment, the user has a guarantee of independent measurement blocks. Note also that the STG may be initialised to a global state drawn from its steady state distribution at the beginning of each experiment.

Table 4.1 Ranges of the mean sojourn times of the states of source types dependent on scale factor [x] and number of sources in a scenario

Number of sources	Lowest range [0]		Highest range [15]	
	min.	max.	min.	max.
1	100 µs	22 s	88 ms	205 hours
2000	100 µs	11 ms	88 ms	6 min

4.2.6 Load extremes

By specifying patterns where all cells are assigned, a 100 % load is obtainable. This is of interest as a debugging feature. For traffic experiments, the maximum load is defined by the user, and may exceed 100 % and cause internal cell losses. The superposition of the traffic from the individual sources in a scenario, Section 2.4, may be regarded as a multiplexer with a limited buffer size. Cell losses, internal to the generator, have no effect on measurement accuracy, as sequence numbering and time stamping etc. is done in the subsequent test cell module. The internal loss is logged and reported at the MMI.

The lowest periodic traffic the user can specify is about three cells per second. This is attained by the use of maximum long patterns and a one state source. Non periodic traffic has no practical lower bound – one cell per fortnight without activation and deactivation.

4.2.7 Activation and deactivation

In the hardware an activation or a deactivation of a source represents a workload similar to a state change. The instantaneous capacity is one activation – deactivation per 12.5 µs. Requests exceeding the instantaneous value will be buffered and the generation of traffic will deviate slightly from what the user has specified.

Due to the operating system, no hard real time requirement can be met for the activation – deactivation. An activation – deactivation operation takes place within 0–250 ms. The delay is very close to the lower bound when the system is lightly loaded. During manual load control, this inaccuracy is less than the timing precision of the operator. A change from 0 to 2047 active sources can be undertaken within a couple of seconds. This is beyond the expected rate of load change at the connection level.

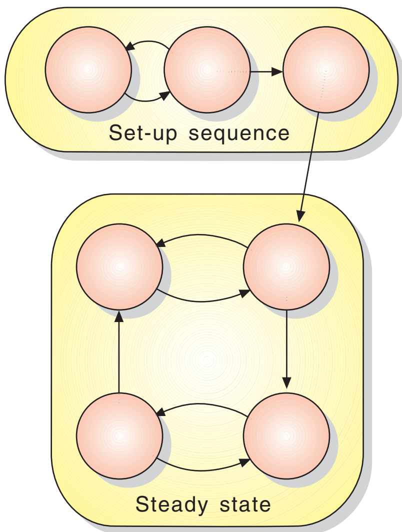


Figure 4.5 Example of state model with a set of initial transient states

4.3 Extended functionality

The sections above have described the current functionality of the STG. This was defined a number of years ago, and advances within the ATM standardisation and measurement technology invite additional functionality. The rest of the section describes such functionality. Some of it is a rather simple task to implement, other parts are quite demanding, and advanced functionality is still at the research stage.

4.3.1 Connection level

The STG was designed before any signalling standard for ATM systems had emerged. Hence, it lacks the possibility to signal with the target system over the ATM link. In the future, the ease-of-use of the generator will increase with the capability of signalling.

If such signalling is going to be implemented, it invites a more advanced connection level. A statistically defined activity on this level, modelling the connects and disconnects of the sources, may be introduced. Furthermore, measurements may be performed on the connect – disconnect activity to determine performance measures like set-up delay, rejection probability, etc.

4.3.2 Basic functionality

A *memory traffic generator,* see Section 2.2.1, can be made as a sub-set of the STG functionality. This will give the user a possibility to configure some of the STGs as a memory based generator with a defined sequence, nearly without any additional hardware. A simultaneous STG and memory based generation is also feasible.

The STG may be used to *introduce delay and cell loss impairments* caused by a finite or infinite buffer in a live cell stream. Hence, an STG may be used to simulate the effect of a connection, e.g. a video transmission, caused by interfering traffic in the network. The interfering traffic is defined in the STG. A feasibility study shows that this functionality is less hardware demanding/costly than the current generation functionality.

Since the STG originally was designed as a module in the PARASOL measurement system very little *measurement functionality* is included. This may, however, be done at a moderate cost. Such functionality comprises:

- Sequence numbering of outgoing cells. This may be done for all VCI/VPIs, or for explicitly defined VCI/VPIs.
- Timestamping of outgoing cells.
- Cell payload integrity control by CRC or other schemes.

4.3.3 Advanced functionality

Cell losses should be extremely *rare events* in ATM systems. An end-to-end loss rate of 10^{-9} is a typical objective. In spite of this, cell losses represent severe service degradations, and measurements of cell losses are important to validate system performance. However, even real time measurements in this range take from days to months. Hence, if QoS requirement in this range shall be investigated, speed up techniques are necessary. Another paper in this issue is devoted to this topic [1]. Ongoing research on introducing such techniques in an STG based measurement set-up is reported in [36]. There are, however, still stones to be turned before such techniques are generally applicable.

In the current version of the STG, all the cells from a source are identical. The user may, however, be interested in measuring specific events related to the source, e.g. start and/or stop of *higher level data units*. For instance:

- Start and stop of an application packet or a transport layer packet
- Start of frame in a video connection.

During the modelling process, such events may be recognised while defining a pattern, cf. Section 3.2. The pattern may be extended to a larger alphabet than {0,1} and these events may be given a unique symbol in the pattern and the correspondingly generated cell tagged accordingly. This will enable delay and loss measurements associated with these higher level data units.

Currently all sources in a scenario have a steady state behaviour. In the STG, the sources may be initialised according to the steady state distribution. Sources are activated in one of these states. Introducing a more advanced connection level as described in Section 4.3.1, it may, however, be of interest to let activated sources run through a separate series of states, e.g. to mimic a *unique set-up behaviour* of the source. This is illustrated in Figure 4.5.

5 Concluding remarks

This paper has hopefully conveyed

- why measurements under realistic load conditions are necessary to be confident in ATM systems, and, in this context, the importance of generation of a realistic load
- the state of the art in ATM source type modelling
- which of the activities of the user, the information source, the network and the end-user equipment that determine the final ATM traffic stream
- how the above activities may be captured in the Composite Source Model
- the principles behind the Synthetic Traffic Generator, its design and capabilities
- necessary and/or possible improvements foreseen.

A range of source type models are under definition according to the principles and methodology described above. The Synthetic Traffic Generator is now being used for traffic measurements and experiments at a number of sites. Among them are the RACE 2 project EXPLOIT, Telia Research, Telenor Research & Development and DELAB. The equipment is also made available for participants in the ACTS programme through the national host in Norway, Telenor AS.

Acknowledgements

The part of the work reported here which has been done at DELAB, is mainly carried out under contracts with Telenor Research & Development. The author would like to express his thanks for support and interest. A number of persons have contributed to the results reported. A simulation study of the CSM buffering characteristics has been carried out by Norvald Stol (DELAB). The design and development of the STG has been a joint effort of a number of persons and companies. The driving force behind this work has been Leif Morland (DELAB). Ole Melteig (Telenor R&D[4]), Steinar Brevik[5] (Telenor R&D), Jens C Michelsen (KTAS[4]), Lars Nedergaard (KTAS[4]) and Per Skovgaard (KTAS[4]) have done a great job in the design specification of the generator and in the hardware development during the PARASOL project. The current software version was developed as a part of a co-operation between Wandel and Golterman and Telenor Research & Development. Thor Gunnar Eskedal (Telenor R&D), Michel Martin (W&G) and Magnar Sunde (DELAB) have contributed to the result. Poul Heegaard (DELAB) has been working on establishing models of specific source types.

References

1 Heegaard, P E. Speed-up techniques for simulation. *Telektronikk,* 91(2), 195–207, 1995. (This issue.)

2 Neuts, M F. *Matrix-geometric solutions in stochastic models.* John Hopkins University Press, 1981.

3 Hemmer, H, Huth, P T. Evaluation of policing functions in ATM networks. I: *Proceedings 13th international teletraffic congress 19–26 June, 1991. Workshops on queuing, performance and control in ATM,* 15, 111–116, 1991.

4 Anick, D, Mitra, D, Sondi, M. Stochastic theory of a data-handling system with multiple sources. *Bell system technical journal,* 61(8), 1871–1894, 1982.

5 Stol, N. *Fluid flow based modelling and analysis of delays in integrated services networks.* Dr. ing. dissertation, Norwegian Institute of Technology, 1995. Draft.

6 Rossiter, M. The switched Poisson process and the SPP/G/1 queue. I: *Proceedings of 12th international teletraffic congress, ITC 12,* 3.1B.3.1–7, June 1988.

7 Malgeris, A et al. Performance models of statistical multiplexing in packet video communication. *IEEE transactions on communications,* 36(7), 834–844, 1988.

8 Li, S-Q, Hwang, C-L. Queue response to input correlation functions : discrete spectral analysis. *IEEE/ACM transactions on networking,* 1(5), 522–533, 1993.

9 Li, S-Q, Hwang, C-L. Queue response to input correlation functions: continous spectral analysis. *IEEE/ACM transactions on networking,* 1(6), 678–692, 1993.

10 Sheng, H-D, Li, S-Q. Queue response to input correlation functions: discrete spectral analysis. *IEEE/ACM transactions on networking,* 1(5), 522–533, 1993.

11 Briem, U, Theimer, T H, Kröner, H. A general discrete-time queueing model : analysis and applications. I: *Tele and data traffic : proceedings from the 13th international teletraffic congress, ITC 13,* 13–19. Elsevier Science Publishers BV, 1991.

12 Virtamo, J T, Roberts, J W. Evaluationg buffer requirements in an ATM multiplexer. I: *The 8th Nordic teletraffic seminar, NTS-8,* IV.4.1–.11. Tekniska Högskolan i Helsingfors, 1989.

13 Ramamurthy, G, Sengupta, B. Modelling and analysis of a variable bitrate video multiplexer. I: *Seventh ITC specialist seminar,* New Jersey, 1990.

14 Heyman, D, Tabtabai, A, Lakshman, T V. Statistical analysis and simulation study of video teleconference traffic. *IEEE transactions circuits systems and video technology,* 2(1), 1992.

15 Grünenfelder, R et al. Measurement and ARMA model of video sources in ATM environment. I: *Tele and data traffic : proceedings from the 13th international teletraffic congress, ITC 13,* 981–985. Elsevier Science Publishers BV, 1991.

16 Reininger, D, Melamed, B, Raychaudhuri, D. Variable bit rate MPEG video characteristics, modeling and multiplexing. I: *Proc. of the*

[4] *Employer when the work was carried out.*

[5] *Steinar Brevik lost his life during the project period.*

14th international teletraffic congress (ITC 14). J. Roberts (ed.). Antibes Juan-les-Pins, June 1994, 295–306. Elsevier, 1994.

17 Leland, W E et al. On the self-similar nature of ethernet traffic. *IEEE/ACM transactions on networking,* 2(1), 1–15, 1994.

18 Norros, I. A simple model for connectionless traffic with long-range correlations. I: *The 11th Nordic teletraffic seminar (NTS-11).* K. Lindberger (ed.). Stockholm, 1993. Telia Research.

19 Duffy, D E et al. Statistical analysis of CCSN/SS7 traffic data from working CCS subnetworks. *IEEE journal selected areas in communications,* 12(3), 544–551, 1994.

20 Arvidsson, Å, Harris, R. Performance comparison of models of individual and merged bursty traffics. I: *The 10th Nordic teletraffic seminar, NTS-10,* Århus, 1992. Jydsk Telefon.

21 Helvik, B E. A model for real-time generation of ATM traffic from a large number of sources. I: *The 9th Nordic teletraffic seminar, NTS-9.* Kjeller, Norwegian Telecom Research, 1990. (Also available as ELAB-RUNIT report STF40 A90162.)

22 Helvik, B E, Hokstad, P, Stol, N. Correlation in ATM traffic streams: some results. I: *Queueing performance and control in ATM.* 25–32. C.D. Pack (ed.). Elsevier, 1991. (Also available as DELAB report STF40-A91110.)

23 Cox, D R, Miller, H D. *The theory of stochastic processes.* Chapman and Hall, 1965.

24 Heegaard, P E, Helvik, B E. Establishing ATM source models for traffic measurements. I: *The 11th Nordic teletraffic seminar (NTS-11),* Stockholm 1993. K. Lindberger (ed.). Telia Research. (Also available as STF40 A93110.)

25 Brady, P T. A model for generating on-off speech patterns in two way conversation. *Bell system technical journal,* 48, 2445–2472, 1969.

26 Heffes, H, Lucantoni, D. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE journal on selected areas in communication,* SAC-4(6), 856–868, 1986.

27 Sriram, K, Whitt, W. Characterizing superposition arrival processes and the performance of multiplexers for voice and data. *IEEE journal on selected areas in communication,* SAC-4(6), 833–846, 1986.

28 Danzig, P B et al. An empirical workload model for driving wide-area TCP/IP network simulations. *Internetworking : research and experience,* 3, 1–26, 1992.

29 Lucantoni, D M, Neuts, M F, Reibman, A R. Methods for performance evaluation of VBR video traffic models. *IEEE/ACM transactions on networking,* 2(2), 176–180, 1994.

30 Osteyee, D. Testing Markov properties of time series. I: *Time series analysis,* 385–400. O.D. Anderson and M.R. Perryman (eds.). North-Holland, 1981.

31 Aagesen, F A. A flow management architecture for B-ISDN. I: Integrated broadband communication Networks and services (IBCN&S), *IFIP Transactions,* C-18, 261–274. V.B. Iversen (ed.). North Holland, 1993.

32 Melteig, O. Introduction to the PARASOL project. I: *The 9th Nordic teletraffic seminar.* Kjeller, Norwegian Telecom Research, 1990.

33 Crawford, G, Grupp, W. Network performance measurements in ATM systems. *Telecommunications,* 22–32, August 1991.

34 Tausworthe, R C. Random number generated by linear recurrence modulo two. *Math. comp.,* 19, 201–209, 1965.

35 Zierler, N, Brillart, J. On primitive trinomials (mod 2). *Information and control,* 13, 541–554, 1968.

36 Helvik, B E, Heegaard, P E. A technique for measuring rare cell losses in ATM systems. I: *Proceedings of the 14th international teletraffic congress (ITC 14),* Antibes Juan-les-Pins, 1994, 917–930. J. Roberts (ed.). Elsevier. (Also available as DELAB report STF4094083.)

Speed-up techniques for simulation

BY POUL E HEEGAARD

Abstract

Due to increased complexity and extreme quality of service requirements, traditional means for performance evaluation of telecom networks do not always suffice. The most flexible approach has been, and still is, simulation, which is the main focus of this paper. However, severe problems regarding efficiency exist, e.g. evaluation of ATM cell loss rate in range of $< 10^{-9}$ which is very computer intensive and hardly feasible within reasonable time with direct simulation. This calls for new and more efficient simulation techniques. An overview of some speed-up approaches is given.

The paper also includes a simple quantitative comparison study of three novel techniques; *importance sampling, RESTART* and *transition splitting*. As an example of practical use, a technique combining three speed-up approaches is described at the end of this paper. It is prepared for both simulation and real-time measurements.

1 Introduction

During recent years we have experienced a rapid evolution of telecom networks with increasingly heavy requirements to the quality of service (QoS). Additionally, the network/system increases in size and complexity, and the technical evolution moves the bounds for operational speed increasingly higher.

The increase in the QoS requirements is a natural consequence of the central role that telecom systems play in today's society. A drastic decrease in the system availability or service quality may have serious or catastrophic consequences for a large number of people. A well known example is the AT&T network breakdown in 1990 where a large part of the subscribers in New York was out of service for nearly 9 hours [1].

The telecom market is now opening for competition on several areas. The customers will be more restrictive to their choice of suppliers, and in some cases require a guarantee of QoS level.

The great challenge in evaluation of these systems is not only the QoS requirements but also the network performance combined with the increasing system complexity, with a distributed nature and a considerable tight logical coupling. Traditionally, three main approaches to evaluation applies. First, the *analytic model* approach which can be very efficient, but it requires a high level of abstraction, considerable effort and skills, and good system knowledge to make a tractable and realistic model. The complexity of telecom systems makes this a formidable and in many cases unattainable task without unrealistic assumptions.

Second, the *simulation model* approach is more flexible than analytic models in the sense that arbitrary level of details are allowed. Computations (simulations) are though normally more demanding, and in systems with heavy QoS requirements and high network performance, the situation is even worse. We have to run through an enormous number of events before we get any observation of interest to the performance measure, and an even larger number to obtain confident and stable estimates. The service degradation (e.g. lost traffic or system failure) is the *rare event* in this context.

The third and direct approach is *measurement* on a network/system. This requires that the system (or at least a prototype) exists, which is a considerable cost and may put strong limitations on the flexibility of setting up the experiment. In a controlled laboratory environment the same limitations does not exists. For example, the B-lab at the Telenor R&D [2] has a test environment where various equipment for the forthcoming ATM technology (Asynchronous Transfer Mode) can be offered synthetic traffic. The traffic generator is the Synthesized Traffic Generator (STG) which is a part of the measurement system developed during the RACE project PARASOL (R1083) [3].

As mentioned, the problem due to rare events in the simulation approach is also a problem in an measurement context, even though the number of events pr. time unit in a measurement is larger than in a simulation experiment. As an example, consider the often stated requirement in ATM switching systems of cell loss ratio in the range 10^{-9} – 10^{-11}. In dependability evaluation the corresponding problem is unavailability requirements of typically less than 10^{-6}. It is well known that to validate such figures, extremely long measurement or simulation periods are required. The average time necessary to observe 20 cell losses in traffic offered a 150 Mbit per second link with a load of 0.65 is about 24 hours – 100 days. Losses tend to occur in bursts, which requires even longer periods for good estimates, and measurement on a single connection makes it even worse. Measurements of other rare events in ATM systems, e.g. the tail of delay and cell jitter distributions, or rare events related to dependability (e.g. TMN or IN) poses the same problem as measuring cell losses.

Hence, the "rare events" represent a serious problem for the dimensioning and evaluation which are of utmost importance to ensure QoS and to avoid expensive overdimensioning. The dimensioning/evaluation run the danger of producing results too late for any practical purpose at an indefensible high cost or produce inaccurate/unreliable results. However, even though the traditional approaches do not apply, there exist a variety of improvements to the simulation approach that aim at reducing the time consumption required for a simulation experiment to obtain accuracy of results. Any method which reduces the simulation run time is referred to as a *speed-up simulation technique*.

This paper discusses work reported in the literature on *speed-up techniques* over the past few years. After a general introduction in Section 2, an overview is found in Section 3-5. A trivial example is also introduced to illustrate and compare the most efficient methods for our purpose. Section 6 presents a framework for measurement of ATM equipment where a combination of speed-up techniques are applied.

2 General on speed-up techniques

A simulation experiment rely on a minimum number of observations, N_{min}, to obtain sufficiently accurate estimates. With restrictions on the maximum measurement period, t_{max} (e.g. due to cost constraints), it turns out that there are experiments that are not frugal according to cost/benefit considerations. In Figure 1 this is illustrated, where the shaded area is the frugal one. Hence, a wide variety of techniques for speeding up simulations are reported in the literature, see e.g. [4,5] for surveys. The following sections give a brief overview along three different approaches: *distributed* and *parallel simulation, hybrid* techniques, and *statistical variance reduction* techniques. The first approach exploits computational parallelism to reduce the "experimental" period by increasing the number of calculations per second. Figure 1 shows how this change in "event rate" can make the simulation experiment a frugal one. The savings in the total measurement period are indicated by the thick shaded dashed line. The latter two approaches are aiming at reducing the number of events necessary to obtain sufficient estimates, i.e. increase the shaded area. This change in lower bounds on number of events may cause a drastic increase in efficiency, see Figure 1 for an example.

The following subsections present some examples of each of the approaches, and their respectively applicability and limitations are discussed. Figure 2 gives a schematic overview of the speed-up techniques that will be discussed in proceeding sections. It is important to notice that the techniques are not all mutual exclusive but may be combined to increase the speed-up gain, see e.g. the measurement technique described in Section 6.

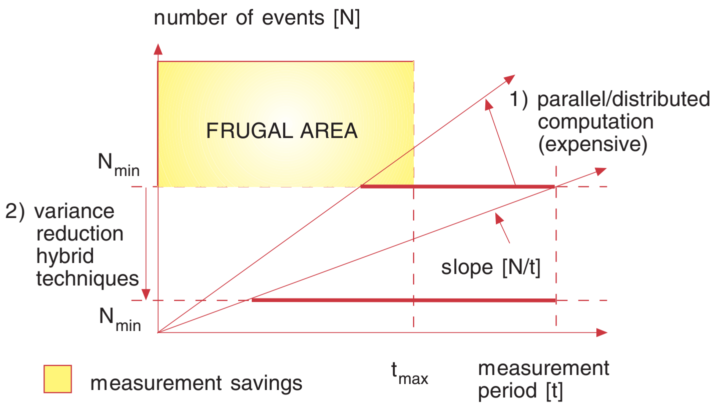


Figure 1 Increasing simulation efficiency by 1) increasing the number of computations per second, or 2) decreasing the minimum number of events necessary for accurate results

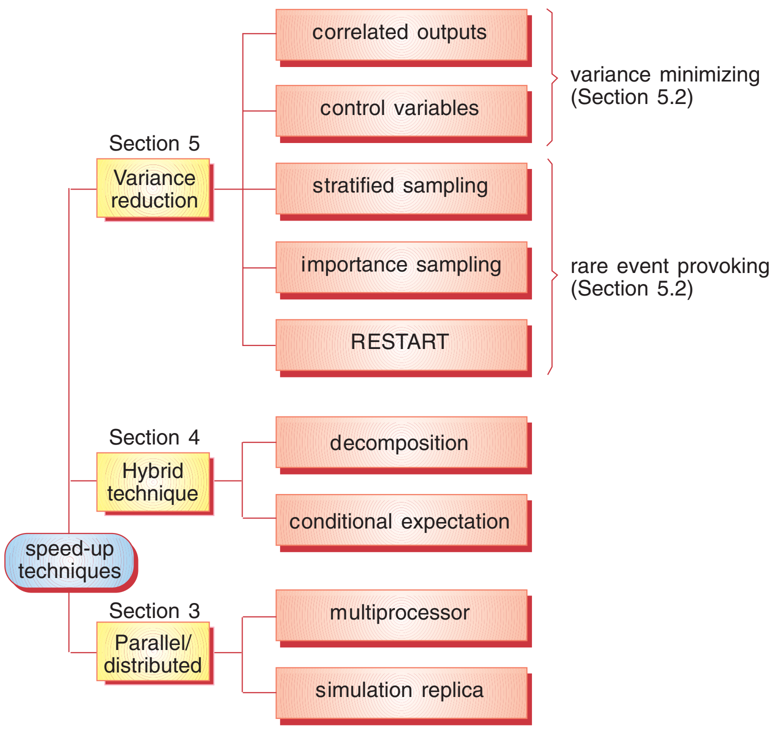


Figure 2 Speed-up technique overview

3 Distributed and parallel simulation

The obvious way of speeding up a simulation is to increase the computational power, i.e. increase the number of events pr. second. This may be done either by using a fast platform such as the CRAY, apply special purpose HW simulation (TASK AC316 in ACTS: "High-speed Networks"), or distribute the simulation experiment on several processors/machines. Unfortunately, this is often rather expensive and, as may be viewed in Figure 1, not always sufficient to obtain accurate results.

Nevertheless, this section comments on two related approaches that aims at reducing the amount of real calendar time needed to conduct the experiment, both by distributing the computational burden on several processors. *Combined with e.g. variance reduction* techniques, see Section 5, you have a powerful simulation.

Distributed and parallel simulation is a field of large interest which have an annual workshop "Parallel and Distributed Simulation (PADS)".

3.1 Multiprocessor simulation

In this approach, the simulation model is partitioned into *processes,* each modelling a physical process of the real system, e.g. a node in a local area network. In a multiprocessor environment (either a multiprocessor machine or network of workstations), each of this processes is executed in parallel on a separate processor. The processes are not independent (if so it is trivial) and hence the interactions must be dealt with. An interaction where both parties must be present at the same place at the same time is a so called *rendezvous synchronisation*. These interactions prevents a K speed-up factor in a K-processor environment because a process must at some occasions stop and wait for input from another process and cannot proceed correctly without it.

Three different approaches to handle interactions are reported in the literature [4]. They handle the synchronisation conflict differently, e.g. how to react when reaching an event in a simulation subprocess where an arrival of a message is expected but not yet arrived.

First, the *optimistic* strategy assumes asynchronous processes where no synchronisation conflicts, e.g. assuming that no message will be sent that ought to be consumed by the current process and the process continues. If future events reveals a conflict, the process must be rolled back to an event previously to the conflict, and the sequence of events must be re-simulated. This strategy is implemented by using the *Time Warp* mechanism [6] or *Uniformation* [7]. The major difference is that Time Warp records all events between the synchronisations, and rolls back to the event previous to the conflict, while Uniformation only records the synchronisation events.

Second, a conservative (or pessimistic) strategy will always wait until a message is available or should have been sent in potential synchronisation conflicts. If no conflict was present, the delay was unnecessary and a waste of time. In [8] this strategy is applied on simulation of the Signalling system no 7, showing increased speed-up factor as the link utilisation increases, see Figure 3. But, the synchronisation overhead will at low utilisation become so high that the distributed simulation will be slower than a single processor run.

As opposed to the event-driven approach from the first and second strategy, a third strategy based on *time-driven* approach is proposed [9]. The global simulation clock is incremented at fixed time intervals rather than on each event. In synchronised network this has shown a significant speed-up in the simulation.

3.2 Parallel simulation replica

As an alternative to use more than one processor on each simulation run, we might as well divide our experiment into K subsections, and run each subsection on K different processors (e.g. workstation) [4,10,11]. This will give a speed-up factor of K.

This is as a much simpler and effective approach than the multiprocessor simulation. The main motivation is that in each simulation experiments you need more than one independent observation period, e.g. using the sectioning method and starting K separate subsections, or in regenerative simulation divide the N regeneration periods into K subset of N/K periods each. You simply make K replica of the simulation (with different seed to the random generator) instead of running a sequence of K multiprocessor simulation where the latter have a speed-up factor less than K.

Not only is this a more simple approach than the previous, it is also less expensive because we may apply general purpose computers, e.g. heterogeneous workstations in a network, instead of costly special purpose hardware.

4 Hybrid techniques

A hybrid simulation technique is generally a technique that combines the flexibility of simulation with the computational efficiency of analytic models. Hybrid techniques have reported substantial computational saving applied on communication networks [5]. The effect is best when a few elements of a network structure must be simulated a number of times.

In this section I will take a closer look on two model dependent approaches that applies to hybrid technique.

4.1 Decomposition

Hybrid technique applies to *hierarchical decomposition* where the system model can be divided into submodels. It is most powerful when the submodels are analytically tractable. The speed-up effect is both due to the effectiveness of analytic solution, and the fact that submodels have to be calculated only once. Hence, even if no analytic solution exists, you still get a speed-up effect.

Both spacial and time domain decomposition can be done. As an example on the latter, consider the ATM end-to-end simulator developed at SINTEF DELAB [12]. This operates on models with large differences in the time scales on different levels, e.g. burst level (100 ms) and cell level (0.01 ms) which means that a large number of cells is sent between each change at burst level. In the simulator it is recently introduced a decomposition speed-up approach where the cell level is simulated until a stationary state is reached. After a change at the burst level (e.g. an increase in the number of sources sending a stream of cells) a period of unstable situation is observed, called a *transient* where e.g. the number of buffered cells increases, and this transient can not be neglected [13]. Hence, the end-to-end simulator simulates this transient period, and then skip the stationary period after a good estimate of the performance in the stationary situation is obtained [14]. Skipping the stationary period may induce considerable computational savings, e.g. for a simple 3-queue example in end-to-end simulator the simulation speed-up factor was in order of 10.

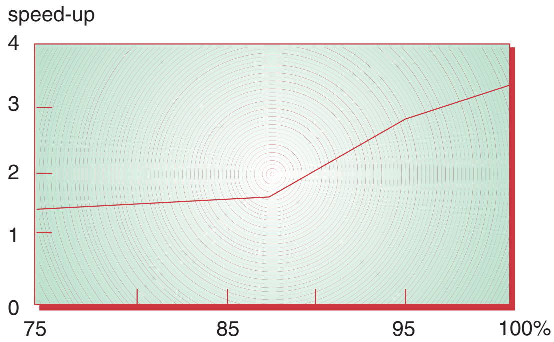


Figure 3 Speed-up versus link utlisation in a 4 processor example, from [2]

Time scale decomposition is also applied in combination with the time warp strategy to induce parallelism [15]. Time scale instead of spatial decomposition is assumed to reduce the overhead, i.e. fewer roll-backs and re-simulations, see Section 3.1.

Application of decomposition is very specific to the system under study, and requires detailed system expertise. Restrictive assumptions that reduce the simulation model flexibility must be put on the models to allow decomposition into interactive submodels.

4.2 Conditional expectation

In decomposition an engineering approach is taken, and the model is broken into functional blocks. As opposed to this, a mathematical approach is taken in *conditional expectation,* where an analytic model is the starting point and the remaining quantities that have no analytic solution are obtained by simulation. Consider e.g. a simple system consisting of 2 servers, where the probabilities of having 1 respectively 2 servers active have an analytic solution. The system response given that 1 and 2 server(s) is active can not be found analytically, and hence must be obtained by simulation. Finally, the mean system response is obtained by combining the probabilities of having 1 and 2 servers, and the simulated responses from the respective cases.

Both decomposition and conditional expectation are model dependent. A disadvantage is that it is generally very difficult to identify the conditioning quantity with an expectation that can be found analytically.

5 Statistical variance reduction

Another approach to speed-up simulation is to use some statistical property of the model to gain variance reduction. These "family" of techniques exploit some statistical properties to reduce the uncertainty in the output estimates and hence lessen the variance. This section presents a handful of variance reduction techniques that are reported in the literature more or less applicable to communication network simulations; *antithetic sampling, common random numbers, control variables, stratified sampling, importance sampling,* and *RESTART*. All, except the latter, are described in [16]. The RESTART method [17,18] is specifically developed for solving rare events in queuing systems like in ATM. The first three belong to a class of methods which exploits dependencies between observations to minimise the variance, while the latter three provoke events of interest by manipulating the underlying process in some way.

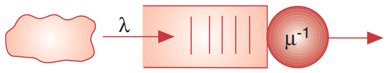
First, to give an indication of the effectiveness of the various methods, a simple reference example is introduced. A comparison study of the provoking event methods is conducted and reported at the end of this section.

5.1 Reference system

A very simple example is introduced to give a relative comparison of the speed-up techniques, and compare them to a direct ("brute-force") simulation where none of the novel statistical variance reduction techniques are applied. The reference system has a well known and simple solution, see e.g. [19], and the simulation study in this article is of illustration purpose only.

It can be interpreted in either of the two following ways:

(i) Queuing system (M/M/1/N-1)

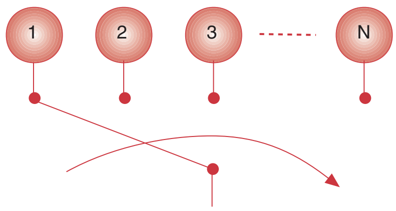


N-1 queueing size

The queuing system have a buffer size of N - 1, which means that up to N customers can be in the system at the same time. The performance measure of interest is the probability of having a full system (i.e. N simultaneous customers). If this probability is very small, it causes problems in evaluation by direct simulation.

If the system is restricted to Poisson arrivals with rate λ and service time following a exponential distribution with mean μ^{-1}, an analytic solution can easily be obtained.

(ii) Highly dependable 1-of-N system



Consider a system of N components where only 1 is active at the time. If this component fails, the system switches over to a non-failed or repaired component, as illustrated in the figure above. Only the active component is sustainable to failure, and only one component can be repaired at the time. The system fails when all components have failed. The probability of a system failure is very low, and causes the same simulation problems as in (i).

The failures are considered as stemming from a Poisson-process with a rate of λ, and the repair times are exponentially distributed with mean μ^{-1}.

A model of these reference systems is described by a *state diagram*, see Figure 4. A state is either (i) the number of customers in the system, or (ii) the number of failed components. The arrows between the states show the possible transitions, with the rate indicated at the top of each arrow.

The performance measure is either (i) the probability of having a full system (N simultaneous customers), or (ii) the probability of N failed components. In either case this is equal to the probability of being in state N and in both cases this probability in question is extremely low.

In Table 1 you find the parameter settings for the 3 cases used in this section, including key results such as the probability of full system, *P(N)*.

5.2 Variance minimising

All variance reduction techniques in this subsection are based on the idea of exploiting dependencies between observations to reduce the resulting variance.

5.2.1 Correlated samples

Two simple approaches try to obtain correlation between the samples by offering complementary input streams to the same model, or the same stream to alternative models. Both antithetic sampling and common random numbers are easy in use, e.g. antithetic sampling is a built-in construct in the DEMOS class [20,21] over SIMULA [22]. Both are exploiting the fact that the variance to a sum of random variables are reduced if they are pairwise negatively correlated.

5.2.1.1 Antithetic sampling

Antithetic sampling performs two complementary input samples to get two negatively correlated outputs.

Table 1 Parameters and key results for the 3 example cases

Case	N	λ	μ	cycle time[a]	P(N)[b]
I	4	0.05	1	21.053	2.9688×10^{-7}
II	6	0.05	1	21.053	7.4219×10^{-10}
III	85	0.8	1	6.25	9.2634×10^{-10}

a The time from leaving state 0 until next time state 0 is left.

b Probability of observing the system in state N.

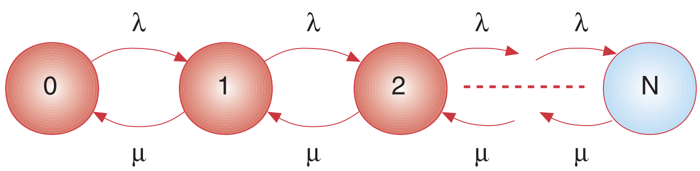


Figure 4 State diagram description of the model of the reference system

To get a picture of how this works consider a series of tosses of a coin where we are interested in the number of heads in 100 experiments. The input to this experiment is a series of numbers drawn from a uniform distribution between 0 and 1. The output is either head or tail. We assume that the output is head if the input value is below a certain value denoted *p*[1] (if we have a fair coin this value is 0.50). The antithetic input is constructed by inverting the value of the original input, e.g. if the original value is 0.32 the antithetic value is 1 - 0.32 = 0.68. This antithetic value is now tested against *p*. If we have a fair coin, i.e. *p* is 0.5, the results in the antithetic case will always be the opposite as in the original case, i.e. a head turn to a coin and vice versa. Hence, the variance reduction is infinite for *p* = 0.5 because the correlation coefficient between the original and antithetic output streams are -1 and the variance is then 0. Figure 5 shows a plot of the ratio between the variance of independent tosses versus antithetic samples. As mentioned, at the probability *p* = 0.5 the theoretical variance of antithetic samples is 0. Note that the speed-up effect decreases rapidly on both sides of *p* = 0.5.

The consequence to the reference example that was introduced in Section 5.1, is that we interchange the 01-variates that determine the sequence of arrivals and services, or failures and repairs. Unfortunately, the correlation coefficient between two antithetic samples in this case is not equal to -1 if the *upward* and *downward* transition rates are different, e.g. the arrival- and service-rates in the M/M/1 queuing example.

Table 2 illustrates the effect of antithetic sampling compare to an experiment of replicated simulation blocks not introducing any correlation. Due to the effect illustrated in Figure 5, the speed-up is reduced as the ratio between upward *(λ)* and downward *(μ)* transition rates decreases. Balanced transition rates, i.e. λ/μ = 1, corresponds to *p* = 0.5.

Antithetic sampling is very easy to use, and in e.g. the SIMULA class DEMOS it is a built-in construct which makes it even more simple. The results in Table 2 were generated by a simple DEMOS-program.

In general, we would not have the strong correlation between output and input

Table 2 Effect of antithetic sampling: Comparison of estimated mean and standard deviation for antithetic sampling versus a number of replications from a direct simulation. The offered load is varied from 0.9 to 0.05

λ/μ	N	antithetic sampling		replication		exact
		mean $\hat{P}(N)$	standard deviation	mean $\hat{P}(N)$	standard deviation	
0.9	4	0.126	1.826 x 10^{-3}	0.125	4.837 x 10^{-3}	0.126
0.5	4	1.578 x 10^{-2}	5.349 x 10^{-4}	1.571 x 10^{-2}	1.273 x 10^{-3}	1.587 x 10^{-2}
0.1	4	9.308 x 10^{-6}	1.053 x 10^{-5}	6.730 x 10^{-6}	1.160 x 10^{-5}	9.000 x 10^{-6}
0.05	4	1.359 x 10^{-7}	3.843 x 10^{-7}	1.359 x 10^{-7}	5.434 x 10^{-7}	2.969 x 10^{-7}

which makes it possible to obtain complementary samples. It is also very difficult to keep two sequence of events synchronised, e.g. as in the case where a head is switched to tail. Lack of synchronisation may produce *positive* correlation, which means that we get an *increase* in the variance instead of a reduction.

Hence, antithetic sampling must be handled with care, and is not recommended in large, complex systems such as network of queues where e.g. the end-to-end delay is of interest.

5.2.1.2 Common random numbers

In many ways this is similar to antithetic sampling. This method is very useful when comparing two (nearly) similar models, e.g. M/M/1[2] and M/D/1 system where the only difference is that the first has exponentially distributed server times while the latter is deterministic. They are offered *common random numbers,* which is the same random numbers sequence. The idea is to produce (nearly) the same sequence of events (e.g. upward- and downward transitions). The comparative performance measure, e.g. the difference in probability of full system, will have reduced variance, if they stay synchronised.

Common random numbers are as easy to use as the antithetic sampling, but at the same time it suffers from the same limitations. You still have the problem of maintaining synchronisation between two parallel simulation experiments, and variance reduction is relying on the dependence between input and output sampling. Hence, the conclusion from Section 5.2.1.1 goes for common random numbers as well.

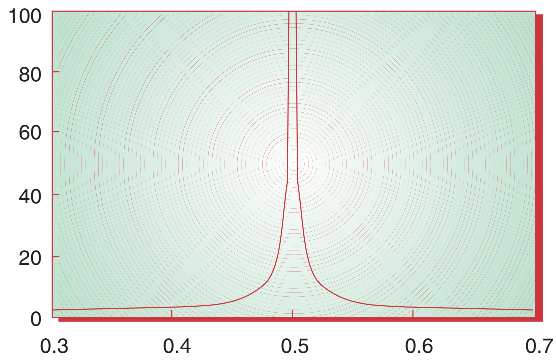


Figure 5 Antithetic sampling: speed-up for 100 tosses of a coin with varying probability of head, p

5.2.2 Control variable

The control variables goes back to a old idea in statistics [16]. Consider the case where you have a set of pairwise observations of the random variables *X* and *C*. You know something about *C*, e.g. the expectation *E(C),* and you want to estimate something about *X*, e.g. the *E(X).* The question is how to exploit the fact that you know *E(C)*? If *X* and *C* are dependent then the answer is *control variables.*

The basic idea is to use the known value *E(C)* as an "anchor" with a rubber chain to pull the expectation of *X* towards its true value. This has the effect of reducing the variance of the observations.

In its most simple form, called *simple control variable,* we just take the mean difference between the observed values of *X* and *C* and claim that it should be equal to the difference between the expected values of *X* and *C*. Rewritten we have the following simple expression for an estimator of *E(X):*

[1] *The p-value is the probability of head.*

[2] *Kendall's notation.*

$$\tilde{X} = X + \beta(C - E(C)) \quad (5.1)$$

where β is -1 in the simple case where X and C are positively correlated and in the same order of magnitude. If not, then β can be picked according to some optimum criteria, see [16] for details on what is called *regression-adjusted* control.

It is useful to distinguish between *internal* and *external* controls:

(i) Internal control

The random variable C with the known expectation occurs in the same computation/simulation as X, e.g. C is the average queuing delay and X is the probability of full system.

(ii) External control

The random variable C with the known expectation occurs in a separate computation/simulation in addition to the one where X is recorded. For example, if we are interested in estimating X which is the probability of full system in a G/G/1[3]-queue, we can claim that it may be correlated to the probability of full system in a M/M/1[4]-queue, and let C be this quantity.

External control has the disadvantage that extra computations are necessary to obtain the control. A general disadvantage with control variables is that it is very difficult to find a proper control variable.

5.3 Rare event provoking

All three methods in these first subsections are means for controlling the variability of a set of observations. It gives us no means for manipulating the processes generating the observations, e.g. to increase the sampling of the rare sequences of event that are of vital importance to the performance measure such as probability of full system in orders less than e.g. 10^{-9}. The next three subsections will present techniques that do.

5.3.1 Stratified sampling

Probably the most frequent use of stratified sampling is in a Gallup poll. The population is divided into strata in accordance to some descriptives such as sex, age, position, education, geographical address, etc. To get a proper impression of the public opinion, a fair sample from each strata must be drawn.

Applied in network performance evaluation, stratified sampling is best explained through an example, details of the method can for instance be found in Chapter 11.5 in [16]. Consider the reference example *M/M/1* from Section 5.1, only this time with k servers, i.e. an *M/M/k*-system. The number of servers vary randomly following a known distribution, i.e. we know the probability of having k servers. Assume that we want to estimate a quantity X that is positively correlated to the number of servers K, which is the *intermediary* variable. We get a reduction in the variance of X if we partition (stratifies) the total calculation into strata according to the possible outcome of the *intermediary* variables, and then estimate X for a given stratum (e.g. a fixed number of servers active).

For example, let k vary from 1 to 4 and suppose we want to estimate the probability of waiting time larger than 5 minutes (which is most likely to occur when only one server is active). We partition the number of simulation replica into 4 strata, and see to that most of the simulation effort are concentrated on the strata containing 1 and 2 servers.

The example illustrates the main problem with stratified sampling, namely how to divide the simulation replica into strata, and how to define the optimal number of simulation replica in each stratum, and finally how to calculate the strata probabilities.

Recently, it is published a variant of stratified sampling, called *transition splitting* [23]. This technique is extremely efficient on models as the reference example, because it uses all available exact knowledge and leaves very little to be simulated. In the M/M/1 case the quantity of interest is known, and therefore the results of Section 5.4 should be very flattering and in favour of transition splitting. The basic idea is simply to assume that in every simulation replica (using *regenerative* simulation, see e.g. [24]) he starts with a sequence of upward transitions from empty until a full system is reached, and then simulate the sequence of events (transitions) from this point and back to an empty system again. The sequence of upward transitions until full system without returning to an empty system first, is the only stratum in this simple example. The probability of this stratum can in this simple case be calculated exact, e.g. by first step analysis [25]. Details on transition splitting can found in [23].

Generally, it is very hard to find a way to do this transition splitting and to define the strata involved. Additionally, it is not a trivial matter to calculate the probability of each stratum. In fact, in our reference example it is more demanding to calculate the stratum probability than to find the exact state probabilities.

5.3.2 Importance sampling

Importance sampling (IS) is a technique for variance reduction in computer simulation where the sampling of the outcome is made in proportion with its relative importance on the result. For a general introduction see for instance [16] Chapter 11.

Importance sampling was originally proposed as a method for *Monte Carlo integration*[5], see e.g. [26]. Later, the method has been applied to several areas. The relevant applications in this paper is assessments within the dependability [27,28] and teletraffic area [29,30,31]. In [32] IS is proposed as a method to provoke cell losses by a synthetic traffic generator like the STG developed at the PARASOL project. The method should be applicable to both real measurement on ATM equipment as well as simulation, see Section 6 and [33].

The basic idea of importance sampling is to change the underlying stochastic process in a way that makes the rare events of interest (i.e. that is important to the performance quantity of interest) to appear more frequently. The observations made under these conditions must of course be corrected in a way that makes them look like being generated by the original process and produce unbiased estimates.

By considering the reference example once more, we soon get an idea of how

[3] *General arrivals and server distributions.*

[4] *Poisson arrival and neg.exp. service time distributions, easy computational, see reference example.*

[5] *Make large number of "shots" at a value domain and count the number of "hits" in the region bounded by the function to be integrated. The ratio of hits to shots is the integration of the bounded region. This is a helpful method in cases where an analytic or numerical solution is difficult or impossible, e.g. due to discontinuity.*

this method works. Remember that the problem of simulating a system containing rare events is that we get a large number of simulation cycles that e.g. does not visit state N. The reason is simply that after leaving state 0 you will most likely return to this state (from state 1) if the difference between the upward and downward transition probabilities are large (in favour of a downward transition). If we manipulate the transition probabilities we can get another behaviour. Increasing the upward relative to downward transition probabilities we increase the frequency of cycles which are visiting state N (the state of interest, e.g. defect or full system probability) before returning to the starting state 0. Hence, the number of simulation cycles needed for observing a full system are reduced. However, these observations are not stemming from the original process and will therefore give biased estimates. Therefore, we have to rescale our observations by the *likelihood ratio,* which is the accumulated ratio between the transition probabilities of the recorded path in the original and the new processes. During the simulation cycle we have to record this ratio in addition to the quantity of interest. In Figure 6 you find an illustration of 4 simulation cycles of the reference example with only $N = 2$.

The results produced by application of importance sampling is reported to be very sensitive to how much the underlying distribution is changed [11,34], e.g. the transition probabilities. To the author's knowledge, optimal biasing only exist for very limited "classes" of systems, such as the reference system. For one-dimensional models like the reference example, you find in the literature application of either

- *balanced biasing,* i.e. the transition probabilities are similar and equal to 0.5, or
- *reversed biasing,* i.e. upward and downward probabilities are interchanged.

In the reference example, the latter is the *optimal* biasing [29,35].

In dependability simulation the change in parameters is concerning an increase in the probabilities of failure, basically not very far from the ideas behind traditional accelerated lifetime testing of HW components, see e.g. [36]. In the literature we find a number of approaches where IS is applied, and studies are done to try to find an optimal, or at least a "good" way

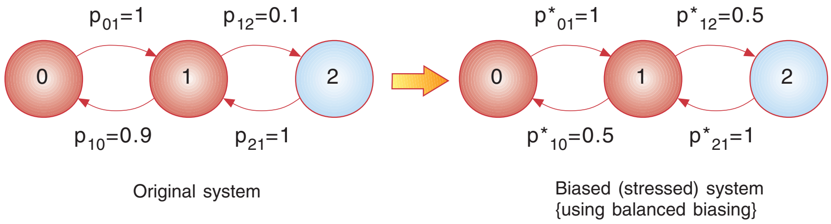


Figure 6 Simulation cycles with importance sampling

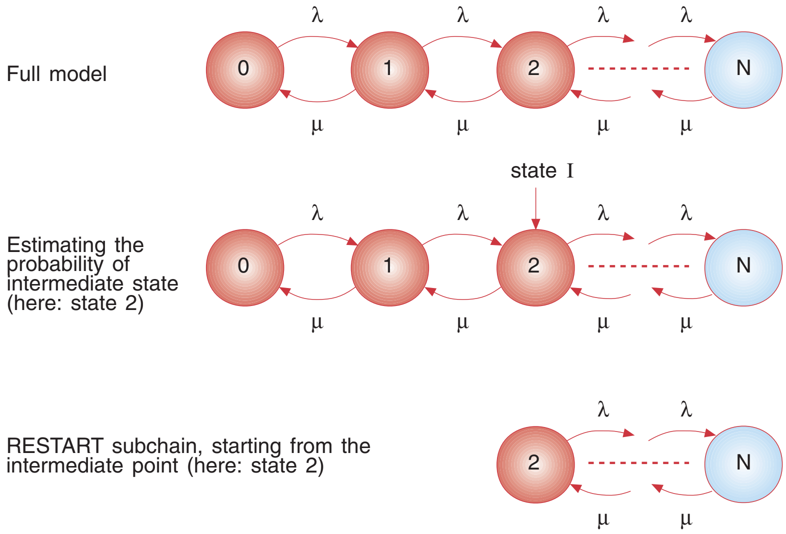


Figure 7 The RESTART method on the reference example

of changing the parameters, such as failure path estimation [34], minimise empirical variance [37], balanced failure biasing [38], failure distance approach where the parameters are optimised during simulations [39].

Similarly, in a teletraffic performance study of e.g. the cell loss rate in an ATM queue, the parameter change is due to an increase in the offered load to the queue.[6] Generally, of course we have the same problems in teletraffic as in dependability of finding an optimal change in parameters. But, in the specific case of our reference example, Parekh and Walrand [29] reported that reversed biasing is an optimal biasing where *Rare event theory* is the theoretical fundament [35]. The results from Section 5.4 show an significant increase in the speed-up gain using reversed instead of balanced biasing.

5.3.3 RESTART

Another importance sampling concept is the RESTART (REpetitive Simulation Trials After Reaching Thresholds), first introduced at ITC'13 by Villén-Altamirano [17]. The basic idea is to sample the rare event from a reduced state space where this event is less rare. The probability of this reduced state space is then estimated by direct simulation, and by Bayes formulae the final estimate is established.

Consider the model of our reference example, the rare events are visits to the state of full system of system failure (state N) before returning to the initial state 0. This means that during a direct simulation experiment, a typical sequence of event is visits to states near the state 0 (the regenerative state). The RESTART reasons as follows: Say that

[6] *Note that the theory of IS restricts us to either increase the load per customer, not the number of customers, or increase the mean service time per server, not to decrease the number of servers.*

Table 3 Case 1: N = 4, λ/μ = 0.05

Method		mean[a] $[10^{-7}]$	S[b] $[10^{-7}]$	CPU-sec (ELC)	efficiency measure[c], $m_i\,[10^{-7}]$	speed-up factor[d]
Direct simulation[e]		11.31	5.984	106.2	635.5	1.0
RESTART[f]	I_{opt} -1 = 1	2.180	1.329	62.5	83.1	7.7
	I_{opt} = 2	3.418	0.804	67.4	54.2	11.7
Transition splitting[f]		2.974	0.00660	35.5	0.23	2,712.3
IS[f]	balanced	2.983	0.07396	29.0	2.1	296.3
	reversed	2.967	0.00825	46.2	0.38	1,667.3

a Exact value 2.968750046 x 10^{-7}
b Standard deviation of the mean estimator
c S*CPU
d Ratio relative to the direct simulation
e No. of regeneration cycles = 200,000 - 7
f No. of regeneration cycles = 20,000

Table 4 Case 2: N = 6, λ/μ = 0.05

Method		mean[a] $[10^{-10}]$	S[b] $[10^{-10}]$	CPU-sec (ELC)	efficiency measure[c], $m_i\,[10^{-10}]$	speed-up factor[d]
Direct simulation[e]		9.048	6.398	52853.1	338,154	1.0
RESTART[f]	I_{opt} - 2 = 1	6.064	2.101	25142.8	52,825	6.4
	I_{opt} - 1 = 2	4.844	2.424	1327.5	3,218	105.1
	I_{opt} = 3	11.554	3.165	1077.0	3,409	99.2
	I_{opt} + 1 = 4	6.888	2.258	25546.7	57,685	5.9
Transition splitting[f]		7.403	0.0163	47.4	0.8	437,672.0
IS[f]	balanced	7.173	46.29	37.5	1,736	194.8
	reversed	7.398	2.875	64.0	184	1,837.8

a Exact value 7.421875000 x 10^{-10}
b Standard deviation of the mean estimator
c S*CPU
d Ratio relative to the direct simulation
e No. of regeneration cycles = 100,000,000
f No. of regeneration cycles = 20,000

Table 5 Case 3: N = 85, λ/μ = 0.8

Method		mean[a] $[10^{-10}]$	S[b] $[10^{-10}]$	CPU-sec (ELC)	efficiency measure[c], $m_i\,[10^{-10}]$	speed-up factor[d]
Direct simulation[e]		<no obs>	-	426261.1	-	-
RESTART[f]	I_{opt} - 1 = 41	2.794	3.645	654.5	2,385.7	1.0
	I_{opt} = 42	2.482	3.358	729.2	2,448.7	1.1
	I_{opt} + 1 = 43	1.427	1.917	697.3	1,336.7	3.4
	I_{opt} + 2 = 44	1.025	1.484	766.8	1,137.9	2.1
	I_{opt} + 3 =45	1.304	1.870	876.6	1,639.2	1.5
Transition splitting[f]		9.298	0.062	3238.8	200.8	11.9
IS[f]	balanced	11.45	3.137	395.9	1,241.9	1.9
	reversed	9.181	0.192	1209.3	232.2	10.3

a Exact value 9.263367173 x 10^{-10}
b Standard deviation of the mean estimator
c S*CPU
d Ratio relative to RESTART with I = 41
e No. of regeneration cycles = 200,000,000
f No. of regeneration cycles = 20,000

we first estimate the probability of being in state *I*, which is visited significantly more often than state *N*. After an accurate estimate for state *I* is obtained, we change the regenerative point of our simulation experiment to state *I* (and allow no visits to states < *I*). Then we get an enormous increase in the number of visits to states far away from state 0, which (hopefully) includes state *N*. Visits to state *N* is more likely when starting from state *I* than from state 0. Figure 7 illustrates how the Markov chain of my reference example is divided into two subchains at the intermediate state *I*.

To find an optimal intermediate state (state *I*) and an optimal number of cycles in either subchain, the use of *limited relative error* (LRE) to control the variability is proposed [40].

The RESTART inventors have expanded their method to a multi-level RESTART [18] by introducing more than one intermediate point, i.e. more than one subchain.

5.4 Comparison of rare event provoking techniques

5.4.1 Results

The simulation results of all 3 cases of Section 5.1 are given in Table 3 to 5 in this section. The results are from one (long) simulation experiment consisting of a (large) number of regenerative cycles.

The columns in the table have the following interpretation:

- The intermediate point in which RESTART splits the state space in two, is denoted *I*, and the theoretically optimal value according to LRE [40], is I_{opt}. I is the intermediate state and the starting state in the RESTART cycles.
- The *mean* value of the observator, i.e. the estimated probability of being in state N, is denoted $\hat{p}_N$.
- The *standard deviation* to this estimate, is denoted *S*.
- The *CPU* time consumption is in seconds, normalised to a SUN 4 sparc station ELC.
- The *efficiency measure* is given as the product of standard deviation *(S)* and the CPU time consumption.
- Speed-up factors are the ratio between the efficiency measure of direct simu-

lation and each method. If no observation are made using direct simulation, the least efficient method is used as the reference level, see Table 5.

The tables include a large number of interesting results, where the main observations are:

- All methods give significant improvement in the computer efficiency compared to direct simulation.
- Transition splitting outranks RESTART and IS in all cases.
- RESTART improves its relative gain to the other methods when the λ/μ ratio and the number of states N in the example are increased.
- The simulation trials with RESTART did not result in optimal gain, measured as the product of empirical mean standard deviation and the CPU time consumption, in the optimal intermediate according to the limited relative error (LRE) method described in [40].
- The optimal biasing strategy, reversed biasing, for importance sampling in our system example, gave significant increase over the often applied balanced biasing, see e.g. [27].

The relative comparison of the empirical optimal RESTART, transition splitting (TS), importance sampling with reversed (IS.rev) and balanced (IS.bal) biasing are found in Tables 6 to 8.

5.4.2 Observations

Several interesting observations were made from the results in Tables 3 to 8. First of all it is important to observe that all three techniques gave stable and unbiased results with significant speed-ups over direct simulation, see [11] for details. In fact, direct simulation is not applicable and it is necessary to invest in an accelerated simulation technique.

From the results it seems obvious that transition splitting (TS) should be chosen because it outranks both RESTART and importance sampling (IS). But, bear in mind that the system example is a rather simple one, which is analytically tractable. The TS heavily depend on the ability to calculate the strata probability. A change in our example, e.g. introducing n different types of customers instead of a single one, makes it no longer tractable. The same happens when the arrival processes is changed to non-Poisson.

Table 6 All-to-all comparison of Case 1 ($N = 4$, $\lambda/\mu = 0.05$)

		Compare			
		RE.optimal[a]	TS	IS.bal	IS.rev
Relative to	RE.optimal	1	231.3	25.3	142.2
	TS	4.3×10^{-3}	1	0.11	0.61
	IS.bal	4.0×10^{-2}	9.2	1	5.6
	IS.rev	7.0×10^{-3}	1.6	0.18	1

a The optimal intermediate point is observed to be $l_{opt} = 2$

Table 7 All-to-all comparison of Case 2 ($N = 6$, $\lambda/\mu = 0.05$)

		Compare			
		RE.optimal[a]	TS	IS.bal	IS.rev
Relative to	RE.optimal	1	4,165	1.9	17.5
	TS	2.4×10^{-4}	1	4.5×10^{-4}	4.2×10^{-3}
	IS.bal	0.54	2,247	1	9.4
	IS.rev	5.7×10^{-2}	238	0.11	1

a The optimal intermediate point is observed to be $l_{opt} = 2$

Table 8 All-to-all comparison of Case 3 ($N = 85$, $\lambda/\mu = 0.8$)

		Compare			
		RE.optimal[a]	TS	IS.bal	IS.rev
Relative to	RE.optimal	1	5.67	0.92	4.90
	TS	0.18	1	0.16	0.86
	IS.bal	1.09	6.18	1	5.35
	IS.rev	0.20	1.16	0.19	1

a The optimal intermediate point is observed to be $l_{opt} = 44$

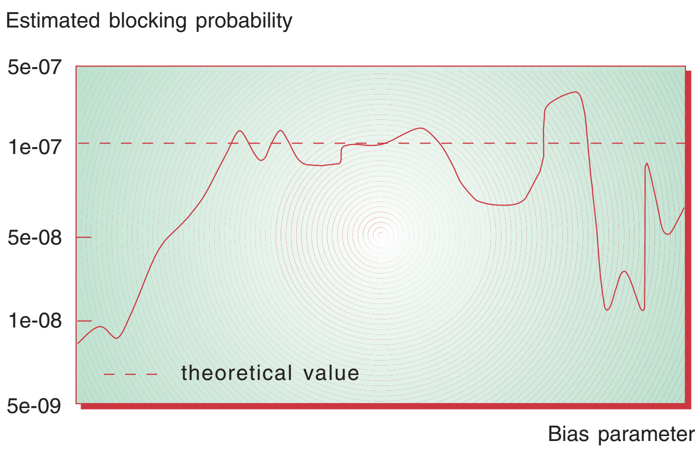


Figure 8 Illustration of how sensitive the estimates are of changing the bias-parameter, figure adopted from [1]

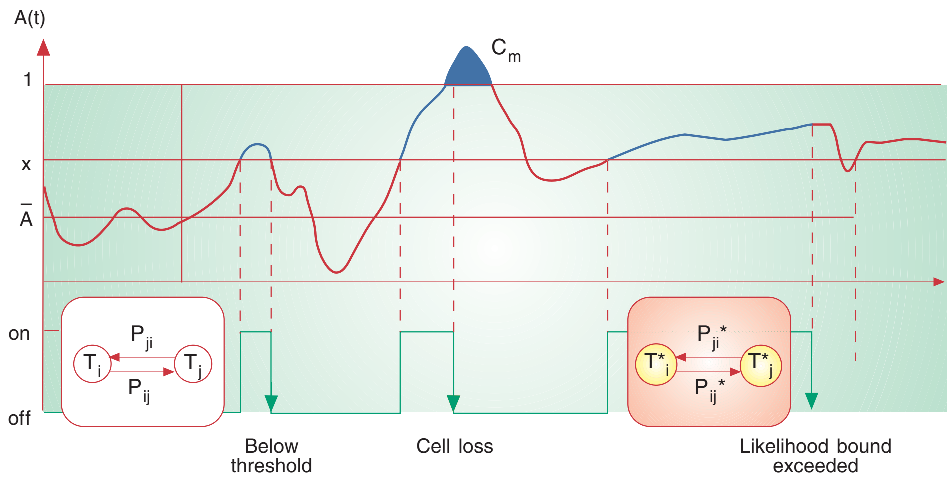


Figure 9 The biasing strategy

Turning to importance sampling (IS) with optimal parameters this seams rather promising. However, although an optimal parameterisation exists in our simple system, this is generally not the case. In fact, if we make the same changes as above, we no longer have a known optimal parameterisation. Nevertheless, even without optimal parameters, IS will provoke rare events rather efficient, but must be handled with care.

Using IS with biasing far from optimal, may produce very wrong estimates, see Figure 8 adopted from [41]. A practical approach to avoid the most drastic effect is reported in [37] where a number of experiments with different biasing parameters are run, and the optimal parameter is found as the one minimising the variance of the likelihood ratio. The results from the comparison study show a difference in gain of 5-10 between the reversed (optimal) and balanced strategy.

The third method, RESTART is not as effective as optimal IS and TS when the l/m ratio is low because the intermediate point might be a rare event as well. The difference is less when the ratio become higher. The multi-level RESTART proposed in [18] is expected to improve this because several intermediate points will be used. However, the effect will probably be most viable where the number of states is large, such as $N = 85$.

RESTART will also experience significant problems in the multi-dimensional state space, this time with defining the intermediate points, i.e. how to reduce the state space to make rare events less rare.

All three methods will experience problems in the multidimensional state space. In all methods the main reason is that we no longer have an easy way of determining the most likely paths (trajectories) in the state space that lead to our rare events of interest.

It is finally worth noting that the three methods are not mutual exclusive. IS may easily be applied both in combination of RESTART and transition splitting. How to combine RESTART and transition splitting is not obvious. We have also gained experience by the use of importance sampling in combination with control variables in a proposed measurement technique for experiments or simulations. This method is described in the following section.

6 Generator based importance sampling

This section presents a generation and measurement technique as an example of practical application of some of the speed-up techniques from previous sections. An abstraction of the problem at hand is shown in Figure 10. The initial objectives was to define a technique that reduced the required measurement period significantly, gave stable and unbiased estimates of rare events like cell losses, and required *no control over the internal state* of the target system. The latter is of utmost importance when it comes to measurement on physical equipment. However, it is anticipated that it is possible to detect the first of one or more cell losses.

The proposed technique that meet these objectives [33] will be outlined in this section. It is based on the *Composite Source Model* which is developed for load generation during controlled traffic measurements with realistic load scenarios [42]. Traffic generation according to this model has successfully implemented in the Synthesized Traffic Generator (STG) [3]. The technique may therefore be implemented in the successors of the STG with moderate changes of the generation hardware to perform measurements on physical systems. It is currently implemented in the DELAB end-to-end traffic simulator [12].

6.1 Approach

The measurement technique consists mainly of three parts. Firstly, the rare event of interest must be provoked by pushing the system into a region where these are more likely to occur. Secondly, the entire measurement period must be defined and divided into independent parts to produce a large number of independent observations for the purpose of statistics. Finally, anchoring the observations and introduce variance reduction are possible by use of internal control variable that can be defined when cell loss is the rare event of interest. The simulation experiments are conducted in parallel on several workstation in network as described in Section 3.2.

So far the work on measurement technique have been devoted to observation of cell losses [33,43] and the presentation in this section will therefore also use cell loss ratio as measurement objective.

6.1.1 Importance sampling

To push the system into a cell loss region we manipulate the load parameters and apply the importance sampling technique, see Section 5.3.2.

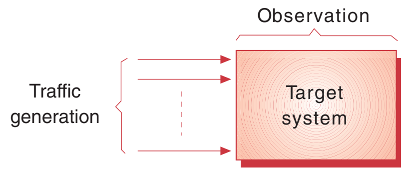


Figure 10 Measurement scenario for application of importance sampling in ATM systems

Table 9 Results from simulation experiments with and without control variables; Number of on-off sources 32 (distributed on 4 generators) Load generated in on-state is 0.04 Source type transition rates are ω_{up} = 1.5 10^{-4} *and* ω_{down} = 2.0 10^{-4} *[cell periods] (Table entries give mean and [95 % confidence intervals])*

Case[a]	No control variable [10^{-7}]	With control variable [10^{-7}]
Switched on at x = 0.72	5.092 [3.901, 6.283]	4.498 [4.427, 4.569]

a A fluid flow approximation of expected cell loss rate is 4.171 x 10^{-7} [44, 45], the true value is expected to be slightly larger due to cell contention contributions

Keep in mind that one of the objectives was to define a measurement technique which have no control of the internal state of the target system. Hence, the importance sampling must be applied to the traffic generation process alone.

The *biasing* is a change in the traffic generator parameters. The biasing strategy switch back and forth between original and biased parameters during an experiment according to the aggregated load profile and a feedback from the target system telling that a cell loss is observed. In Figure 9, you find a typical load profile for the aggregated load from all active sources in the traffic generator process.

The on-off biasing strategy is as follows:

Turn on stressing. When the (aggregated) load level exceeds a certain reference level, *x*, we change the load parameters, i.e. probability of a source becoming active is increased and so is the mean duration of an active source.

Turn off stressing. Three different criteria are defined to turn off stressing (switch back to the original load parameters):

- *cell loss observed:* we have reached our objective.
- *load is below reference level:* our strive to push the system towards a cell loss region was not successful.
- *likelihood ratios exceeds limits:* if the ratio between the original and stressed transition probabilities become to small (or to large) it is an indication of that the current sample is not among the most likely of the unlikely rare event samples.

6.1.2 Load cycles

The next important task is to divide the experiment into independent measurement periods.

The first and obvious choice is to start a number of experiments of fixed duration, called blocked measures. But, it turns out that the observation are extremely sensitive to this fixed block duration; too long period produces unstable results, and too short periods return too few observations (different from 0). No obvious way of defining the optimal length was identified.

We know that regenerative simulation (as applied in the reference example) has been successfully combined with importance sampling for dependability studies, [27, 28]. But, once again remember that we have no control of the internal state of the system, hence no global regeneration point may be found. Regeneration points may be identified in the generator. However, under realistic load conditions they occur so rarely that they are of no practical use.

Instead of trying to identify regenerative points in the entire system we will observe the aggregated load profile and define a regenerative load point at every instance where the load profile exceeds a certain reference level. A load cycle is then the time between two subsequent upward transitions through the reference level. Figure 11 illustrate how a load profile is divided into load cycles separated by $I_0^{(i)}$.

These load cycles are our measurement period. The reference level is chosen as the same level as the where the biasing is turned on and off, but it need not be.

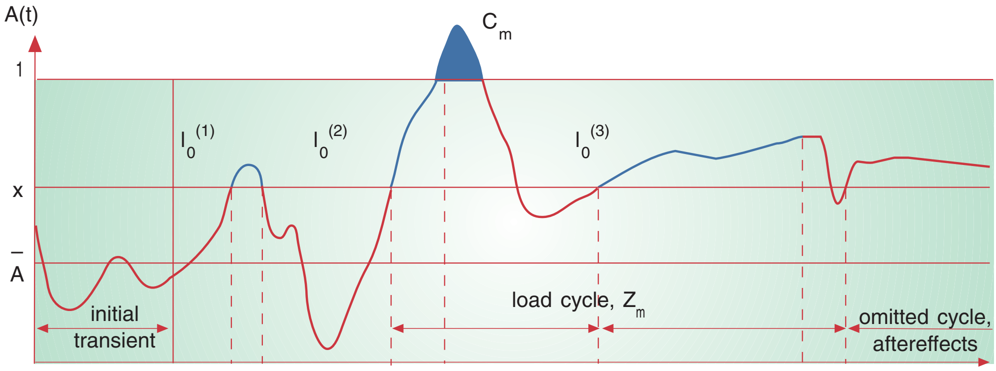


Figure 11 Defining load cycles as measurement period

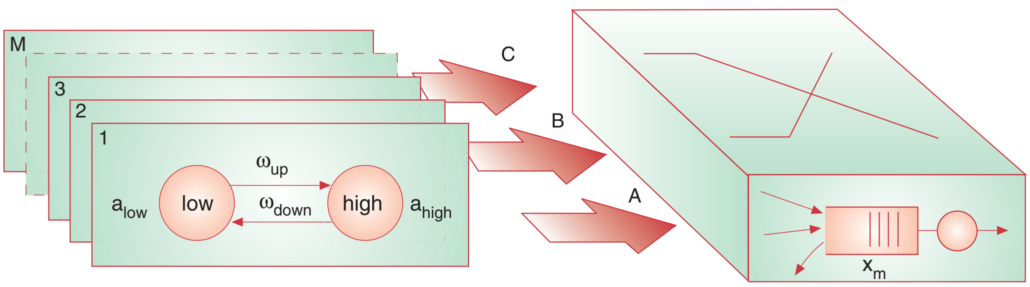


Figure 12 Illustration of the studied load scenario

6.1.3 Control variables

The number of offered cells exceeding the capacity, denoted C_m in Figure 11, is correlated to the number of lost cells which may be recorded in the same simulation experiment and the expectation can be analytically obtained. The correlation is strong if the cell contention contribution is low.

Hence, the C_m is applicable as an internal control variable, see Section 5.2.2, and must therefore be recorded during the measurement along with the number of cell losses.

6.2 Example

Applying the measurement technique on the set-up in Figure 12, with 4 generators of 8 on-off sources each, producing the results of Table 9. The effect of the control variable is significant. Unfortunately, no cell losses were observed in the unstressed experiments and hence the speed-up relative to direct simulation can not be established in this case.

6.3 Remaining tasks

Remaining tasks is further development of the biasing strategy for more complex sources than the on-off sources, application to a heterogeneous scenario (i.e. sources of different traffic properties), and to obtain end-to-end statistics in a queuing networks.

Another interesting task is implementation of this technique in the Synthesized Traffic Generator (STG).

7 Conclusion

The experience in literature and also documented in the study of this article, show clearly that some speed-up technique is necessary in evaluation of the traffic and dependability aspects of the current and future telecom systems and network. Currently, it is significant ongoing research activity on this topic, and with still increasing interest, e.g. within ACTS.

The techniques are not yet mature and ready for practical applications. They are characterised as

- being technically challenging
- containing many nasty pitfalls (we have been into some of them).

Currently, rare event provoking techniques seem most efficient with respect to produce significant speed-ups in simulation. The importance sampling is the most flexible and most promising regarding multi-dimentional state spaces. However, care must be taken in choosing biasing strategy and parameters otherwise very wrong estimates can be obtained.

Running replications of the simulation experiment in parallel is strongly recommended and easy to carry out if a network of workstations is available.

As an example of practical use, several speed-up techniques:

- parallel simulation replicas
- importance sampling
- control variables

were successfully combined in a proposed measurement/simulation technique for testing of ATM-equipment as presented in this paper.

Acknowledgement

I would like to thank my supervisor Prof. Bjarne E. Helvik for fruitful and stimulating discussions and for constructive comments during preparations of this article.

Section 6 is a joint work of Helvik and myself under contract with Telenor Research and Development.

References

1 Fitzgerald, K. Vulnerability exposed in AT&T's 9-hour glitch. *The institute,* 14(3), 1, 1990. A news supplement to IEEE Spectrum.

2 Ormhaug, T. *Plans for a broadband laboratory at Norwegian Telecom Research.* Internal document F91/u/304, -305, Norwegian Telecom Research, 1991. (In Norwegian.)

3 Helvik, B E, Melteig, O, Morland, L. The synthesized traffic generator; objectives, design and capabilities. I: *Integrated broadband communication networks and services (IBCN&S).* IFIP, Elsevier, April 20-23 1993. Copenhagen. (Also available as STF40 A93077.)

4 Kurose, J F, Mouftah, H T. Computer-aided modeling, analysis, and design of communication networks. *IEEE journal on selected areas in communications,* 6(1), 130–145, 1988.

5 Frost, V S, Larue, W, Shanmugan, K S. Efficient techniques for the simulation of computer communications networks. *IEEE journal on selected areas in communications,* 6(1), 146–157, 1988.

6 Jefferson, D R. Virtual time. *ACM transaction on programming languages and systems,* 7(3), 404–425, 1985.

7 Heidelberger, P, Nicola, D M. Parallel simulation of markovian queuing networks. I: *Proceedings of the second international workshop on modelling, analysis, and simulation of computer and telecommunication systems (MASCOT'94).* IEEE Computer Society Press, January 31 – February 2, 35–36, 1994.

8 Kalantery, N et al. Fast parallel simulation of SS7 in telecommunication networks. I: *Proceedings of the first international workshop on modelling, analysis, and simulation of computer and telecommunication systems (MASCOT'93).* IEEE Computer Society Press, 171–175, 1993.

9 Lubachevsky, B, Ramakrishnan, K. Parallel time-driven simulation of a network using a shared memory MIMD computer. I: *Modelling tools and techniques for performance analysis.* D Potier (ed.). North-Holland, 1985.

10 Lin, Y-B. Parallel independent replicated simulation on networks of workstations. I: *Proceedings of the 8th workshop on parallel and distributed simulation (PADS'94).* IEEE Computer Society Press, July 6-8, 71–81, 1994.

11 Heegaard, P E, Helvik, B E. *A technique for measuring cell losses in ATM systems combining control variables and importance sampling.* Trondheim, SINTEF DELAB, 1992. (Technical Report STF40 A92138.)

12 Stol, N, Helvik, B E. *End-to-end traffic in ATM networks : network chain simulator.* Trondheim, ELAB-RUNIT, 1991. (Technical Report STF40 A91035.)

13 Stol, N, Helvik, B E. Simulation of systems with time between events on extremely different time scales: an example. I: *The 8th Nordic teletraffic seminar, NTS-8,* Tekniska Högskolan i Helsingfors, Aug. 29–31 1989, VI.4.1–.17. (Also available as STF40 A89161, in Norwegian.)

14 Stol, N, Helvik, B E. *Simulator enhancements.* Technical memo 40-NO940345, October 1994.

15 Ammar, H H, Deng, S. Time warp simulation using time scale decomposition. *ACM transaction on modeling and computer simulation,* 2(2), 158–177, 1992.

16 Lewis, P A W, Orav, E J. *Simulation methodology for statisticians, operation analysts and engineers, vol. I.* Wadsworth & Brooks/Cole Advanced Books & Software, 1988.

17 Villén-Altamirano, M, Villén-Altamirano, J. RESTART : a method for accelerating rare event simulation. I: *Queueing performance and control in ATM.* C.D. Pack (ed.). Elsevier, 1991, 71–76.

18 Villén-Altamirano M et al. Enhancement of the accelerated simulation method RESTART by considering multiple thresholds. I: *The 14th International Teletraffic Congress (ITC'14).* J. Labatoulle and J. Roberts (eds.). Antibes Juan-les-Pins, France, June 6–10 1994. Elsevier, 797–810.

19 Kleinrock, L. *Queueing systems, theory, vol. I.* John Wiley, 1975.

20 Birtwistle, G. *DEMOS : a system for discrete event simulation on Simula.* MacMillian ed., 1979.

21 Sol, H, Ijeplaar, D. *Expansion of the DEMOS-package with facilities for replications.* Memorandum.

22 Birtwistle, G et al. *SIMULA begin.* Lund, Studentlitteratur, 1979.

23 Gavioronski, A A. Transition splitting for estimation of rare events with application to high speed data networks. I: *The 14th International Teletraffic Congress (ITC'14).* J. Labatoulle and J. Roberts (eds.). Antibes Juan-les-Pins, France, June 6–10 1994. Elsevier, 767–776.

24 Lavenberg, S. *Computer performance modeling handbook.* Academic Press, 1983.

25 Taylor, H M, Karlin, S. *An introduction to stochastic modeling.* Academic Press, 1984.

26 Rubinstein, R Y. *Simulation and the Monte-Carlo method.* New York, Wiley, 1981.

27 Conway, A E, Goyal, A. Monte Carlo simulation of computer system availability/reliability models. I: *Digest of paper, FTCS-17 – The seventeenth international symposium on fault-tolerant computing,* July 6–8 1987, 230–235.

28 Nicola, V F et al. Fast simulation of dependability models with general failure, repair and maintenance processes. I: *Proceedings 20th international symposium on fault-tolerant computing,* 491–498, 1990.

29 Parekh, S, Walrand, J. Quick simulation of excessive backlogs in networks of queues. *IEEE transactions on auto. control,* 34(1), 54–66, 1986.

30 Kesidis, G, Walrand, J. A review of quick simulation methods for queues. I: *International workshop on modeling, analysis and simulation of computer and telecommunication systems (MASCOTS'93),* San Diego, California, January 17–20 1993.

31 Norros, I, Virtamo, J. Importance sampling simulation studies on the discrete time nD/D/1 queue. I: *The 8th Nordic teletraffic seminar (NTS-8).* Tekniska Högskolan i Helsingfors, Aug. 29–31 1989, VI.3.1–.16.

32 Helvik, B E. *Technology for measurement of rare events in ATM systems.* Trondheim, SINTEF DELAB, Working Paper, 1991. (AN91130.)

33 Helvik, B E, Heegaard. P E. A technique for measuring rare cell losses in ATM systems. I: *The 14th International Teletraffic Congress (ITC'14).* J. Labatoulle and J. Roberts (eds.). Antibes Juan-les-Pins, France, June 6–10 1994. Elsevier, 917–930.

34 Geist, R M, Smotherman, M K. Ultrahigh reliability estimates through simulation. I: *Annual reliability and maintainability symposium,* 350–355, Jan. 1989.

35 Bucklew, J A. *Large deviation techniques in decision, simulation, and estimation.* Wiley, 1990.

36 Mann, N, Schafer, R, Singpurwalla, N. *Methods for statistical analysis of reliability and life data.* John Wiley & Sons, 1974.

37 Devetsikiotis, M, Townsend, J K. Statistical optimization of dynamic importance sampling parameters for efficient simulation of communication networks. *IEEE/ACM transactions on networking,* 1(3), 293–305, 1993.

38 Goyal, A et al. A unified framework for simulating markovian models for highly systems. *IEEE transaction on computers,* 41(1), 36–51, 1992.

39 Carrasco, J A. Failure distance based simulation of repairable fault-tolerant computer systems. I: *Proceedings of the fifth international conference on computer performance evaluation : modelling techniques and tools.* G. Balbo and G. Serazzi (eds.). North-Holland, Feb. 15-17 1991, 351–365.

40 Schreiber, F, Görg, C. Rare event simulation : a modified RESTART-method using the LRE-algorithm. I: *The 14th International Teletraffic Congress (ITC'14).* J. Labatoulle and J. Roberts (eds.). Antibes Juan-les-Pins, France, June 6–10 1994. Elsevier, 787–796.

41 Wang, Q, Frost, V S. Efficient estimation of cell blocking probability for ATM systems. *IEEE/ACM transactions on networking,* 1(2), 230–235, 1993.

42 Helvik, B E. A model for real-time generation of ATM traffic from a large number of sources. I: *The 9th Nordic teletraffic seminar (NTS-9).* Kjeller, Norwegian Telecom Research, Aug. 21–23 1990. (Also available as ELAB-RUNIT report STF40 A90162.)

43 Heegaard, P E, Helvik, B E. *Provoking and measuring cell losses in ATM systems by importance sampling, control variables and load cycles.* Trondheim, SINTEF DELAB, 1993. (SINTEF DELAB report STF40 A93148.)

44 Anick, D, Mitra, D, Sondi, M. Stochastic theory of a data-handling system with multiple sources. *Bell system technical journal,* 61(8), 1871–1894, 1982.

45 Østerbø, O, Hemmer, H. Private communication. November 1993.

46 *The 14th International Teletraffic Congress (ITC'14).* J. Labatoulle and J. Roberts (eds.). Antibes Juan-les-Pins, France, June 6–10 1994. Elsevier.

Some important queuing models for ATM

BY OLAV ØSTERBØ

Abstract

In this paper we consider two queuing models of an ATM multiplexer. The first is a discrete time multiserver queuing model with batch arrivals. The second is a single server queuing model where the multiplexer is fed by the superposition of a periodic stream and a batch arrival process. Both infinite and finite buffers are considered and approximate formulae for the cell loss probability are given.

1 Introduction

The future public ATM network will support a variety of services ranging from voice and video to highly bursty and unpredictable data services. To be able to operate such a network properly it is important to have robust methods for dimensioning of network resources.

To be able to meet the QoS demand, and still maintain high utilisation of network resources, it seems to be a good choice to separate delay sensitive traffic such as voice and video from loss sensitive, such as bursty data traffic. In this paper we mainly address the problem of dimensioning of buffer space and links for delay sensitive traffic. For bursty data traffic one needs huge buffers to absorb long data packets, and thereby obtaining statistical multiplexing gain and avoiding peak rate allocation. For these traffic types the burst scale behaviour will be dominant. We refer to [1] as a comprehensive reference of cell scale queuing models for ATM.

To study network performance for the delay sensitive type traffic we propose two different models of an ATM multiplexer. The first model is a multiserver queue in discrete time with batch arrivals where we focus on the queue length distribution and cell loss probability. In the second model we merge a periodical source and a batch arrival stream to see how this affects the performance. Similar models are reported in the literature where two cell streams are offered to a multiplexer, see [2] and [3] for references.

2 A multiserver queuing model for ATM

In the first part of the paper we consider a queuing model for a multiplexer in an ATM network where the ratio between the output and the input lines speed is given by a fixed number (see Figure 1).

The time axis is slotted with each slot equal to a cell transmission time (on the input lines). This means that in one slot on the input lines a fixed number of cells can be transmitted to the output line. So this mode is then actually a multiserver queuing model with deterministic service times. If we assume that the cells arrive independently on each input line and also assume independent arrivals in each slot, we obtain a batch arrival process to the multiplexer. We assume that the batch size follows a general distribution with distribution $a(i) = Pr$ (batch size $= i$) and generating function $A(z)$, and we denote n the speed-up factor. For infinite buffers these models are well described in the literature, see [4] and [5] for references.

The offered traffic of the multiplexer is $A = A'(1)$ and the load

$$\rho = \frac{A}{n}.$$

2.1 Infinite buffer case

To analyse the queuing model described above we use standard z-transform method for a multiserver queue with deterministic service times. We observe the queue size at the end of each slot, and we define the following stochastic variables:

Q_k - the queue length at the end of the k-th slot

A_k - the number of arrivals from the batch arrival process during the k-th slot

The queue length process is then described by the following recurrence equation:

$$Q_k = max[Q_{k-1} + A_k - n, 0]; \quad k = 1,2,... \tag{2.1}$$

We consider the queuing system in the steady state regime, which means that the load is less than the unity, and we define the queue length distribution

$$q(i) = \lim_{k\to\infty} P(Q_k = i),$$

$$Q(i) = \lim_{k\to\infty} P(Q_k \le i)$$

and the generating function

$$Q(z) = \lim_{k\to\infty} E\left[z^{Q_k}\right]$$

2.1.1 Queue length distribution

By applying the z-transform to equation (2.1) we get the following expression for the for the queue length transform:

$$Q(z) = \frac{z-1}{z^n - A(z)}\left[\sum_{j=0}^{n-1} p_j z^j\right] \tag{2.2}$$

where

$$P_j = \sum_{i=0}^{j} p_i$$

and

$$p_i = \sum_{i_1 - i_2 = i} q(i_1)a(i_2)$$

is the probability that there are i cells in the system (queue and service) at the end of a slot.

The unknown constants P_j are determined by the usual function theoretical arguments. This is done by examining the zeros of the function $f(z) = z^n - A(z)$ inside the unit circle in the complex plane. It can be shown that $f(z)$ has exactly n zeros $\{r_1 = 1, r_2, ..., r_n\}$ inside the unit circle $|z| \le 1$. This allows us to obtain the following set of equations to determine the P_j's:

$$\sum_{j=0}^{n-1} P_j = n - A \tag{2.3}$$

$$\sum_{j=0}^{n-1} P_j r_k^j = 0; \quad k = 2, \ldots, n \tag{2.4}$$

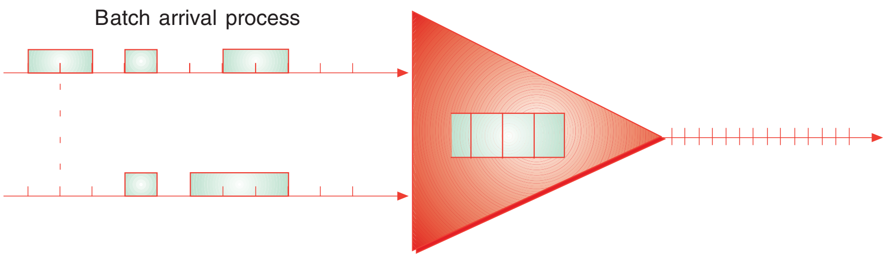


Figure 1 The first queuing model for the ATM multiplexer

The equations (2.3) and (2.4) are of so called "Vandermonde" type, and one simple way of obtaining the solution is done by using a generating function. If we consider the polynomial

$$\sum_{j=0}^{n-1} P_j x^j$$

then (2.3) simply says that r_k is a root in this polynomial. Whence

$$\sum_{j=0}^{n-1} P_j x^j = C \prod_{k=2}^{n} (x - r_k).$$

The unknown constant C is determined by setting $x = 1$, giving

$$C = \frac{n - A}{\prod_{k=2}^{n}(1 - r_k)},$$

so the polynomial generating the P_j's is given by:

$$\sum_{j=0}^{n-1} P_j x^j = (n - A) \frac{\prod_{k=2}^{n}(x - r_k)}{\prod_{k=2}^{n}(1 - r_k)} \quad (2.5)$$

By inverting (2.5) we finally get the following expressions for the coefficients:

$$P_j = (n - A) \frac{(-1)^{n-j-1} \sum_{2 \le i_1 < i_2 \cdots < i_{n-j-1} \le n} r_{i_1} r_{i_2} \cdots r_{i_{n-j-1}}}{\prod_{k=2}^{n}(1 - r_k)}$$

$$j = 0,...,n - 1 \quad (2.6)$$

The form of the queue-length distrbution can be found by inverting the z-transform (2.2):

$$Q(i) = \sum_{j=0}^{\min[1, n-1]} P_j \Theta(i - j) \quad (2.7)$$

where $\Phi(i)$ is given by

$$\Theta(i) = \frac{1}{i!} \frac{d^i}{dz^i} \left\{ \frac{1}{A(z) - z^n} \right\}_{z=0} = \sum_{l=0}^{\lfloor \frac{i}{n} \rfloor} a_{i-nl}^{-(l+1)} \quad (2.8)$$

where a_i^{-k} is the i-th coefficient in the expansion of $A(z)^{-k}$, i.e.

$$a_i^{-k} = \frac{1}{i!} \frac{d^i}{dz^i} \left\{ A(z)^{-k} \right\}_{z=0} \quad (2.9)$$

For some important arrival distributions such as *Poisson (M), Bernoulli (Geo)* and *generalised negative binomial (Nb)* the coefficients above are easily obtained. We have put these results in Appendix A.

Despite (2.7)–(2.9) give explicit expressions for the queue length distributions (at least for *Poisson, Bernoulli* and *generalised negative binomial* arrival processes), these formulae are not quite effective to perform numerical calculations (except for the case $n = 1$). This is due to the fact that the function $\Phi(i)$ defined by (2.8) contains all the roots of $z^n - A(z)$ also with modulus less than the unity, and therefore this function will grow as powers of the inverse of these roots (as function of i). So when i becomes large the expression for $\Phi(i)$ and therefore $Q(i)$ becomes numerically unstable.

One way to overcome this problem is to cancel these roots directly as is done in the z-transform (2.2) (by the boundary equations (2.3) and (2.4)). To do so we also need the roots of $z^n - A(z)$ outside the unit circle. We denote $\{r_{n+1}, r_{n+2}, ...\}$ these roots (which in principle can be infinite if $a(i)$ is infinite). We also assume that all the roots are distinct. By making partial expansions of the z-transform (2.2), we get:

$$q(i) = \sum_{l=n+1} c(r_l) \left(\frac{1}{r_l} \right)^i \quad (2.10)$$

where

$$c(r_l) = \frac{1}{r_l} \lim_{z \to r_l} (r_l - z) Q(z) = \frac{(n - A)(1 - r_l)}{[nA(r_l) - r_l A'(r_l)]} \frac{\prod_{k=2}^{n}(r_l - r_k)}{\prod_{k=2}^{n}(1 - r_k)} \quad (2.11)$$

It can be shown that the root with the smallest modulus say r_{n+1} (outside the unit circle) is real and it is obvious that this root will be the dominant part in the expression (2.10) for large i. The queue size distribution will therefore have the following asymptotic form:

$$q(i) \sim c(r_{n+1}) \left(\frac{1}{r_{n+1}} \right)^i \quad (2.12)$$

for large i.

For most cases of practical interest (2.12) gives satisfactory accuracy and can be used to estimate the cell loss probability by summing (2.12) above the buffer capacity.

2.1.2 Some further analysis of the multiserver queuing model

It is possible to obtain different expressions for the z-transform and the queue length distribution in the previous section which do not involve the roots of $z^n - A(z)$. In Appendix B it is shown that the logarithmic sum:

$$\sum_{k=2}^{n} \log(z - r_k) = \log \frac{z^n - A(z)}{z - 1} + \frac{1}{2\pi i} \int_{|\zeta|=r} \frac{\log\left(1 - \frac{A(\zeta)}{\zeta^n}\right)}{z - \zeta} d\zeta \quad (2.13)$$

valid for $|z| < r$ and

$$\sum_{k=2}^{n} \log(z - r_k) = \log \frac{z^n}{z - 1} + \frac{1}{2\pi i} \int_{|\zeta|=r} \frac{\log\left(1 - \frac{A(\zeta)}{\zeta^n}\right)}{z - \zeta} d\zeta \quad (2.14)$$

which is valid for $|z| > r$ and r chosen (arbitrarily) such that $1 < r < r_{n+1}$.

Taking the limit $z \to 1$ in (2.13) gives us:

$$\sum_{k=2}^{n} \log(1 - r_k) = \log(n - A) + \frac{1}{2\pi i} \int_{|\zeta|=r} \frac{\log\left(1 - \frac{A(\zeta)}{\zeta^n}\right)}{1 - \zeta} d\zeta \quad (2.15)$$

Combining the result above we can write the queue length transform on the following compact form:

$$Q(z) = exp(I(z) - I(1)) \text{ where} \quad (2.16)$$

$$I(z) = \frac{1}{2\pi i} \int_{|\zeta|=r} \frac{\log\left(1 - \frac{A(\zeta)}{\zeta^n}\right)}{z - \zeta} d\zeta \quad (2.17)$$

valid for $|z| < r$

There are now some different ways of using (2.17). First we expand the logarithm in series and integrate term by term. On $|\zeta| = r$ we have $|A(\zeta)| \le A(r) < r^n = |\zeta|^n$ so we have

$$\left|\frac{A(\zeta)}{\zeta^n}\right| < 1.$$

Expanding gives:

$$I(z) = \sum_{k=1}^{\infty} \frac{1}{k} I_k(z)$$

with

$$I_k(z) = \frac{1}{2\pi i}\int_{|\zeta|=r} \frac{A(\zeta)^k}{(z-\zeta)\zeta^{nk}} d\zeta \quad (2.18)$$

The $I_k(z)$ is evaluated by residue calculations at $\zeta = 0$ and $\zeta = z$:

$$I_k(z) = \frac{A(z)^k}{z^{nk}} - \sum_{i=0}^{nk-1} a_i^k z^{i-nk} = \sum_{i=nk}^{\infty} a_i^k z^{i-nk} \quad (2.19)$$

with

$$a_i^k = \frac{1}{i!}\frac{d^i}{dz^i}\left\{A(z)^k\right\}_{z=0} \quad (2.20)$$

Collecting the results above we get the following z-transform for $I(z)$:

$$I(z) = \sum_{i=0}^{\infty} \beta_i z^i \quad (2.21)$$

where

$$\beta_i = \sum_{k=1}^{\infty} \frac{1}{k} a_{i+nk}^k \quad (2.22)$$

For some important arrival distributions such as *Poisson (M), Bernoulli (Geo)* and *generalised negative binomial (Nb)* the coefficients a_i^k are easily obtained. We have put these results in Appendix A.

A different expression may be obtained by taking the derivatives of (2.17) giving

$$\beta_i = -\frac{1}{2\pi i}\int_{|\zeta|=r} \zeta^{-(i+1)} \log\left(1 - \frac{A(\zeta)}{\zeta^n}\right) d\zeta \quad (2.23)$$

For numerical calculations both (2.22) and (2.23) are useful, however, in cases where the coefficients a_i^k are difficult to obtain (2.23) seem to be the most appropriate choice.

Inverting the transform (2.16) gives us the queue length distribution in terms of the β_i's:

$$q_0 = \exp\left(-\sum_{i=1}^{\infty}\beta_i\right) \quad (2.24)$$

and

$$q_i = q_0 \sum_{k_1+2k_2+\cdots+ik_i=i} \frac{(\beta_1)^{k_1}}{k_1!} \frac{(\beta_2)^{k_2}}{k_2!} \cdots \frac{(\beta_i)^{k_1}}{k_i!} \quad (2.25)$$

Although the formulae above seems to be difficult to use for large i, we have nice recurrence formulas to obtain the q_i's:

We define the two-dimensional array γ_i^j by setting $\gamma_0^0 = 1$ and $\gamma_0^j = 0$, $j = 1,...,m$

and we then calculate for $i = 1,...,m$:

$$\gamma_i^j = \frac{1}{i}\left\{(j+1)\gamma_{i-1}^{j+1} + \sum_{l=0}^{j}(l+1)\beta_{l+1}\gamma_{i-1}^{j-l}\right\}$$

for $j = 0,...,m-i$ (2.26)

The desired distribution is then simply $q_i = q_0\gamma_i^0$ for $i = 1,...,m$.

Other performance measures such as the moments of the queue length distribution are easily obtained from the integral representation of the z-transform (2.16) and (2.17). For the mean and the variance we get

E[Q] = I'(1) and
Var[Q] = I''(1) + I'(1) where (2.27)

$$I'(1) = -\frac{1}{2\pi i}\int_{|\zeta|=r} \frac{\log\left(1 - \frac{A(\zeta)}{\zeta^n}\right)}{(1-\zeta)^2} d\zeta = \sum_{i=1}^{\infty} i\beta_i \quad (2.28)$$

and

$$I''(1) = \frac{2}{2\pi i}\int_{|\zeta|=r} \frac{\log\left(1 - \frac{A(\zeta)}{\zeta^n}\right)}{(1-\zeta)^3} d\zeta = \sum_{i=2}^{\infty} i(i-1)\beta_i \quad (2.29)$$

It is also possible to derive asymptotic approximations for large n applying the saddle-point method to the integral (2.17). (See [6] for a detailed outline of the saddle-point method.) We find:

$$I(z) = \frac{c(r_M)}{1 - \frac{z}{r_M}} \quad (2.30)$$

with

$$c(r_M) = \frac{-\log\left(1 - \frac{A(r_M)}{r_M^n}\right)}{r_M\sqrt{2\pi g''(r_M)}} \quad (2.31)$$

where

$g(z) = \log A(z) - n\log z$ is the oscillating part of (2.17) and

$$g'(z) = \frac{A'(z)}{A(z)} - \frac{n}{z} \quad (2.32)$$

and

$$g''(z) = \frac{A''(z)}{A(z)} - \left(\frac{A'(z)}{A(z)}\right)^2 + \frac{n}{z^2} \quad (2.33)$$

Also r_M is chosen so that $g'(r_M) = 0$. It is easy to see that $1 < r_M < r_{n+1}$, and also that $g''(r_M) > 0$ and $c(r_M) > 0$.

By inverting the approximate transform we obtain the following asymptotic queue length distribution for large n:

$$q_0 = \exp - \frac{c(r_M)}{r_M - 1} \quad (2.34)$$

and

$$q_i = q_0\left(\frac{1}{r_M}\right)^i \left\{\sum_{l=1}^{i}\frac{1}{l!}\binom{i-1}{i-l}[c(r_M)]^l\right\};$$

$i = 1,2,...$ (2.35)

2.2 Finite buffer case

To get accurate results for the cell loss probability we must consider models with finite buffers. In this case the govern equation takes the form

$$Q_k = min[M, max[Q_{k-1} + A_k - n, 0]] \quad (2.36)$$

where M is the buffer capacity.

By using (2.36) we derive the following set of equations for the distribution of the number of cells in slot $q(j)$:

$$q(0) = \sum_{i_1+i_2 \le n} a(i_1)q(i_2)$$

$$q(j) = \sum_{i_1+i_2=j+n} a(i_1)q(i_2);\ j = 1,...,M-1$$

$$q(M) = \sum_{i_1+i_2 \ge M+n} a(i_1)q(i_2) \quad (2.37)$$

where $q(i) = 0$ for $i < 0$ and $i > M$.

The finite buffer model may be analysed using an iterative approach by solving the governing equations (2.37) numerically. However, we take advantage of the analysis in section 2.1 and we apply the same method as for the infinite buffer case. We introduce the generating functions

$$Q^M(z) = \sum_{i=0}^{M} q(i) z^i,$$

and by using (2.37) we get:

$$Q^M(z) = \frac{z-1}{z^n - A(z)}$$

$$\left[\sum_{j=0}^{n-1} P_j^M z^j - z^{M+n} \sum_{s=1}^{\infty} \frac{1-z^s}{1-z} g^M(s)\right] \tag{2.38}$$

where

$$P_j^M = \sum_{i=0}^{j} p_i^M$$

and

$$p_i^M = \sum_{i_1+i_2=i} q(i_1) a(i_2)$$

is the probability that there are i cells in the system at the end of a slot, and is the probability that exactly s cells are lost during a slot.

By examining (2.38) we see that for the major part of the queue-length distribution is of the same form as for the infinite buffer case:

$$Q^M(i) = \sum_{j=0}^{\min[i,n-1]} P_j^M \Theta(i-j) \tag{2.39}$$

The coefficients P_j^M's are determined by the fact that (2.39) gives the queue length distribution also for $i = M,...,M+n\text{-}1$, and whence:

$$\sum_{j=0}^{n-1} P_j^M \Theta(M+k-j) = 1;\ k = 0,1,...,n-1 \tag{2.40}$$

The n equations (2.40) are non-singular and determine the P_j^M's uniquely. For small buffers (2.40) is well suited to calculate the coefficients. However, to avoid numerical instability for larger M we expand (2.40) by taking partial expansion of the function Φ. After some algebra we get:

$$\sum_{j=0}^{n-1} P_j^M \left[r_l^j + \sum_{s=n+1} r_s^j A(r_l, r_s)\left(\frac{r_l}{r_s}\right)^M\right]$$

$$= \delta_{1,l}(n-A) \tag{2.41}$$

where

$$A(r_l, r_s) = \left(\frac{nA(r_l) - r_l A'(r_l)}{nA(r_s) - r_s A'(r_s)}\right)$$

$$\left[\frac{\prod_{k=1,k\neq s}^{n}\left(r_s^{-1} - r_k^{-1}\right)}{\prod_{k=1,k\neq l}^{n}\left(r_l^{-1} - r_k^{-1}\right)}\right] \tag{2.42}$$

The generating function for the P_j^M's in this case may be expressed as:

$$\sum_{j=0}^{n-1} P_j^M x^j$$

$$= (n-A)E_1(I+W)^{-1}V(x) \tag{2.43}$$

where E_1 is the n dimensional row vector $E_1 = (1,0,....,0)$, and $W = (W(i,l))$ is the $n \times n$ matrix:

$$W(i,l) = \sum_{s=n+1}\left(\frac{r_l}{r_s}\right)^{M+n-1}$$

$$\left(\frac{nA(r_l) - r_l A'(r_l)}{nA(r_s) - r_s A'(r_s)}\right) V_l(r_s) V_i(r_s) \tag{2.44}$$

and $V(x)$ is the n dimension column vector $V(x) = (V_l(x))$:

$$V_l(x) = \frac{\prod_{k=1,k\neq l}^{n}(x - r_k)}{\prod_{k=1,k\neq l}^{n}(r_l - r_k)} \tag{2.45}$$

The form of (2.41) is useful to determine the overall cell loss probability. Inserting $z = 1$ in (2.38) we obtain the mean volume of cells lost in a slot;

$$E[V_L] = \sum_{s=1}^{\infty} s g^M(s) :$$

$$E[V_L] = \sum_{j=0}^{n-1} P_j^M - (n-A) \tag{2.46}$$

The volume lost can also be determined by direct arguments since we have that the lost volume of cells in a slot equals the offered load minus the carried load i.e.

$$E[V_L] = A - \left(\sum_{j=0}^{n-1} j p_j^M + \sum_{j=n}^{M} n p_j^M\right)$$

which equals (2.46).

From (2.41) taking $l = 1$ we get the cell loss probability

$$p_L = \frac{E[V_L]}{A}$$

as:

$$p_L = \frac{-1}{A}\sum_{j=0}^{n-1}\sum_{s=n+1} P_j^M r_s^j A(1, r_s)\left(\frac{1}{r_s}\right)^M \tag{2.47}$$

evaluating we get

$$p_L = -\frac{(n-A)}{A} E_1 (I+W)^{-1} W E_1^T \tag{2.48}$$

For large M we expand $(I + W)^{-1}$ in (2.48), and by taking only the first term in the expression for W we get the following simple approximate formula for the cell loss probability:

$$p_L \approx \frac{(n-A)^2}{A[r_{n+1}A'(r_{n+1}) - nA(r_{n+1})]}$$

$$\left(\frac{1}{r_{n+1}}\right)^{M+n-1} P^2 \tag{2.49}$$

where P is the product

$$P = \frac{\prod_{k=2}^{n}(r_{n+1} - r_k)}{\prod_{k=2}^{n}(1 - r_k)} \tag{2.50}$$

By using (2.14) and (2.15) we obtain the following integral expression for P:

$$P = \frac{1}{(n-A)}\frac{(r_{n+1})^n}{(r_{n+1} - 1)}\exp(I) \tag{2.51}$$

where I is the integral

$$I = \frac{1}{2\pi i}\int_{|\zeta|=r}\frac{(r_{n+1} - 1)}{(r_{n+1} - \zeta)(\zeta - 1)}$$

$$\log\left(1 - \frac{A(\zeta)}{\zeta^n}\right) dz \tag{2.52}$$

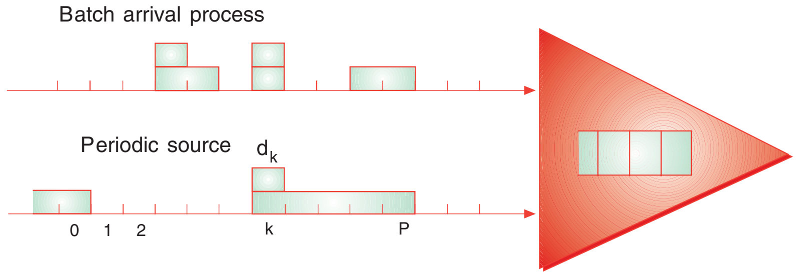


Figure 2 The second queuing model for the ATM multiplexer

The main reason for giving (2.52) is to use the saddle-point method for large n to obtain asymptotic approximations. We get:

$$I \approx \frac{1}{\sqrt{2\pi g''(r_M)}} \frac{(r_{n+1}-1)}{(r_{n+1}-r_M)(r_M-1)} \log\left(1 - \frac{A(r_M)}{r_M^n}\right) \qquad (2.53)$$

where r_M and $g''(r_M)$ are given in (2.31) – (2.33).

3 A queuing model with a periodic source and a background stream

As a second queuing model, we consider a model for a multiplexer in an ATM network with two arrival streams. The arrival process is formed by superposing a periodic stream and a batch arrival process (e.g. Poisson or Bernoulli process). It is assumed that the batch size B_k in each slot is independent and follows a general distribution with distribution $b(i) = P(B_k = i)$ and generating function $B(z)$. The periodic stream is characterised by the (periodic) sequence $\{d_k\}^P_{k=1}$ where d_k is the number of cells arriving from the periodic stream in slot k $(= 1,...,P)$ and P is the period (see Figure 2). For infinite buffers similar models are described in [1], [2] and [3].

The mean load on the multiplexer from the periodic source,

$$\rho_D = \frac{D}{P}$$

with

$$D = \sum_{i=1}^{P} d_i,$$

is the number of cells arriving in a period divided by the length of the period. The load from the batch arrival process is $\rho_B = B'(1)$, so the total offered load on the multiplexer is $\rho = \rho_B + \rho_D$.

3.1 Infinite buffer case

We observe the queue size at the end of each slot, and we define the following stochastic variables:

- Q_k - the queue length at the end of the k-th slot
- B_k - the number of arrivals from the batch arrival process during the k-th slot
- d_k - the number of arrivals from the periodic stream during the k-th slot.

The fundamental law for queuing processes with deterministic service times implies the following recurrence equation:

$$Q_k = \max[Q_{k-1} + B_k + d_k - 1, 0]; \quad k = 1,2,... \qquad (3.1)$$

We consider the queuing system in the steady state regime, which means that the average load over one period must be less than the unity. In this case the distribution of Q_k and Q_{k+iP} $(k = 0,1,...,P - 1)$ is identical, and we define the distributions $q_k(i) = P(Q_k = i)$, $Q_k(i) = P(Q_k \le i)$ and the generating functions $Q_k(z) = E[z^{Q_k}]$; $k = 0,1,....,P$-1.

3.1.1 Queue length distribution

By applying (3.1) we get the following recurrence equations to determine the generating functions:

$$Q_k(z) = Q_{k-1}(z) z^{d_k - 1} B(z) + \left(1 - \frac{1}{z}\right) q_k; \quad k = 1, ..., P \qquad (3.2)$$

where the q_k is the probability that the system is empty at the end of slot k. Solving for $Q_{k+P}(z)$ from (3.2) and using the fact that $Q_{k+P}(z) = Q_k(z)$ we get:

$$Q_k(z) = \frac{z-1}{z^K - B(z)^P} \sum_{j=k+1}^{P+k} q_j z^{-U(k,j)} B(z)^{P+k-j} \qquad (3.3)$$

where

$$U(k,j) = (D_j - D_k) - (j - k - 1) \qquad (3.4)$$

is the unfinished work entering the queue (from the periodic source) during the slots k+1,k+2,...,j, and

$$D_j = \sum_{i=1}^{j} d_i,$$

$D = D_P$ and $K = P - D$. Due to the periodicity we obviously have $q_j = q_{j-P}$, $j > P$.

As for the multiserver case the unknown constants q_j can be determined by function theoretical arguments. This is done by examining the zeros of the function $f(z) = z^K - B(z)^P$ inside the unit circle in the complex plane. It can be shown that $f(z)$ has exactly K zeros $\{r_1 = 1, r_2,....,r_K\}$ inside the unit circle $|z| \le 1$. This allows us to obtain the following set of equations to determine the q_j's:

$$D_j = \sum_{i=1}^{j} d_i, \qquad (3.5)$$

$$\sum_{j=1}^{P} q_j r_k^{j-1-D_j} B(r_k)^{-j} = 0; \quad k = 2,....,K \qquad (3.6)$$

It can be shown that D of the q_j's are zero, so (3.5) and (3.6) determine the other q_j's uniquely. To get the slot numbers for which the q_j's are zero we proceed as follows:

We consider the busy/idle periods for the periodic source, (which of course also will be periodic). If j lies in a busy period (for the periodic source) then obviously $q_j = 0$, otherwise j lies in an idle period and hence $q_j \ne 0$. The same result can also be obtained by considering the leading terms in the z-transform for $Q_k(z)$; which is

$$\sum_{j=k+1} q_j z^{-U(k,j)} O(1).$$

If j lies in busy period (for the periodic source) we can find a k value that gives $U(k,j) > 0$, i.e. gives negative exponent in the z-transform, which implies $q_j = 0$.

For some cases of special interest the boundary equations (3.5) and (3.6) can be solved explicitly. This is the case if the period P contains only one single busy/idle period (for the periodic source). We choose the numbering of the time axis such that $D_j = 0; j = 1,...,K$ (busy period), and we have $q_j = 0; j = K+1,...,P$ (idle period). In this case the boundary equations will be of "Vandermonde" type:

$$\sum_{j=1}^{K} q_j = P(1-\rho) \tag{3.7}$$

$$\sum_{j=1}^{K} q_j (\zeta_k)^{j-1} = 0; \quad k = 2, \ldots, K \tag{3.8}$$

where $\zeta_k = r_k/B(r_k)$.

These equations are identical with (2.3) and (2.4) in section 2.1.1, so the polynomial generating the q_j's may be written as:

$$\sum_{j=1}^{K} q_j x^{j-1} = P(1-\rho) \frac{\prod_{k=2}^{K}(x-\zeta_k)}{\prod_{k=2}^{K}(1-\zeta_k)} \tag{3.9}$$

The coefficients q_j in this case are easily found by the relation between the roots and the coefficients in a polynomial.

The form of the queue-length distributions can be found by inverting the z-transforms (3.3):

$$Q_k(i) = \sum_{j=k+1}^{P+k} q_j \Phi$$

$$(i + U(k,j), j - k) \tag{3.10}$$

where $U(k,j)$ is the unfinished work entering the queuing system from the periodic source given by (3.4) and $\Phi(i,j)$ is given by

$$\Phi(i,j) = \frac{1}{i!}\frac{d^i}{dz^i}\left\{\frac{B(z)^{-j}}{1 - z^K B(z)^{-P}}\right\}_{z=0}$$

$$= \sum_{l=0}^{\lfloor \frac{i}{K} \rfloor} b_{i-Kl}^{-(lP+j)} \tag{3.11}$$

where b_i^{-k} is the i-th coefficient in the expansion of $B(z)^{-k}$, i.e.

$$b_i^{-k} = \frac{1}{i!}\frac{d^i}{dz^i}\left\{B(z)^{-k}\right\}_{z=0} \tag{3.12}$$

For *Poisson (M), Bernoulli (Geo)* and *generalised negative binomial (Nb)* the coefficients above are given in Appendix A, (where the offered A is set to ρ_B).

As for the multiserver model, the explicit expressions for the queue length distributions are not effective to perform numerical calculations for large i. This is due to the fact that the function $\Phi(i,j)$ defined by (3.11) contains all the roots of $z^K - B(z)^P$ also with modulus less than the unity, and hence this function will grow as power of the inverse of these roots. So when i becomes large the expression for $Q_k(i)$ becomes numerically unstable.

As for the multiserver model one can overcome this problem by cancelling these roots directly as it is done in the z-transform (3.3) (by the boundary equations (3.5) and (3.6)). To do so we also need the roots of $z^K - B(z)^P$ outside the unit circle. We denote $\{r_{K+1}, r_{K+2}, ...\}$ as these roots (which in principle can be infinite if $b(i)$ is infinite). We also assume that all the roots are distinct. By making partial expansions of the z-transforms (3.3) we get:

$$q_k(i) = \sum_{l=K+1} c_k(r_l)\left(\frac{1}{r_l}\right)^i \tag{3.13}$$

where

$$c_k(r_l) = \frac{1}{r_l}\lim_{z\to r_l}(r_l - z)Q_k(z)$$

$$= F(r_l)H_k(r_l) \tag{3.14}$$

with

$$F(z) = \frac{1-z}{[K - PzB'(z)/B(z)]} \tag{3.15}$$

and

$$H_k(z) = \sum_{j=k+1}^{P+k} q_j z^{-U(k,j)} B(z)^{k-j} \tag{3.16}$$

In the general case the series for $q_k(i)$ given by (3.13)–(3.16) will not be convergent for $i < K = P - D$. For numerical calculations it is therefore necessary to use (3.10) for small i, (at least for $i < K$) and then the partial expansion (3.13) for large i ($i \geq K$). The first term in the series for $q_k(i)$ (given by (3.13)–(3.16)) is the dominant part for large i, and it can be shown that this root (with the smallest modulus outside the unit circle) is real.

3.2 Finite buffer case

The finite buffer case is analysed by Heiss [7] using an iterative approach by solving the governing equations numerically. However, we take advantage of the analysis in section 3.1 and we apply the same method as for the infinite buffer case.

In this case the governing equation takes the form

$$Q_k = \min[M, \max[Q_{k-1} + B_k + d_k - 1, 0]] \tag{3.17}$$

where M is the buffer capacity.

By using (3.17) we derive the following set of equations for the distribution of the number of cells in slot k (=1,...,P):

$$q_k(0) = \sum_{i_1+i_2+d_k \leq 1} b(i_1)q_{k-1}(i_2)$$

$$q_k(j) = \sum_{i_1+i_2+d_k = j+1} b(i_1)q_{k-1}(i_2);$$

$$j = 1, ..., M-1 \tag{3.18}$$

$$q_k(M) = \sum_{i_1+i_2+d_k \geq M+1} b(i_1)q_{k-1}(i_2)$$

where $q_k(i) = 0$ for $i < 0$ and $i > M$.

Also for the finite buffer case we introduce the generating functions

$$Q_k^M(z) = \sum_{i=0}^{M} q_k(i)z^i.$$

By using (3.18) and solving for $Q_k^M(z)$; $k = 0,...,P-1$ we get:

$$Q_k^M(z) = \frac{z-1}{z^K - B(z)^P}$$

$$\sum_{j=k+1}^{P+k}\left\{q_j^M - z^{M+1}\sum_{s=1}^{\infty}\frac{1-z^s}{1-z}g_j^M(s)\right\}$$

$$z^{-U(k,j)}B(z)^{P+k-j} \tag{3.19}$$

where

$$q_j^M = \delta_{0,d_j} b(0) q_{j-1}(0)$$

is the probability that the system is empty at the end of slot j, and

$$g_j^M(s) = \sum_{i_1+i_2+d_k = s+M+1} b(i_1)q_{j-1}(i_2)$$

is the probability that exactly s cells are lost during slot j. (Due to the periodicity we obviously have

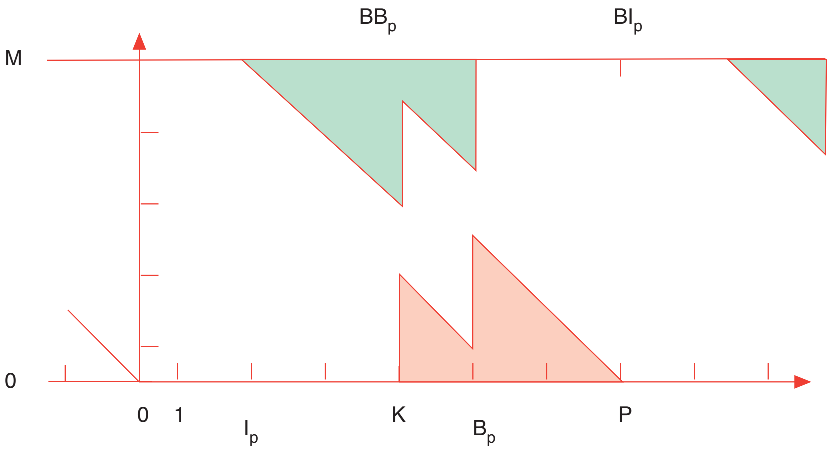


Figure 3 The busy periods and backward busy periods for the periodic source

$q_j^M = q_{j-P}^M$ and $g_j^M(s) = g_{j-P}^M(s)$.

For the sake of simplicity we assume $M \ge u_j$; $j = 1,\dots,P$ where u_j is the workload from the periodic source and is given by

$$u_j = \max\{U(k,j);\ k = j-1, j-2 \dots\} \quad (3.20)$$

(If $M < u_j$ for some j one may redefine the periodic stream by taking $d_j^\dagger = d_j - (u_j - M)$ if $u_j > M$ and $d_j^\dagger = d_j$ if $u_j \le M$. In this case the workload from the periodic source may be calculated by $u_j = \max[\min[u_{j-1}, M] - 1, 0] + d_j$. For the redefined system $M \ge u_j^\dagger$ will be fulfilled, however, one must count for the extra losses $d_j - d_j^\dagger$.)

By examining (3.19) we see that for the major part of the queue-length distribution is of the same form as for the infinite buffer case:

$$Q_k^M(i) = \sum_{j=k+1}^{P+k} q_j^M \Phi(i + U(k,j), j-k) \quad (3.21)$$

which is valid for $i = 0,1,\dots,M-w_k$ (the white area in figure 3) where

$$w_k = \max\{U(k,j);\ j = k+1, k+2, \dots \quad (3.22)$$

The values k for which $w_k > 0$ constitute what we call the backward busy periods (BB_p), and the values for which $w_k = 0$ give the backward idle periods (BI_p) (see Figure 3).

As for the infinite case we have $q_j^M = 0$ when j lies in a busy period (for the periodic source). The rest of the q_j^M's are determined by the fact that (3.21) gives the total queue length distribution for the k values in the backward idle periods, and hence:

$$\sum_{j=k+1}^{P+k} q_j^M \Phi(M + U(k,j), j-k) = 1 \quad (3.23)$$

The K equations (3.23) are non-singular and determine the non zero q_j^M's uniquely. The rest of the $Q_k^M(i)$'s (for $i = M-w_{k+1},\dots,M$ and k lies in a backward busy period; the shaded area in the figure) may be calculated using the state equations (3.18).

For small buffers (3.23) is well suited to calculate the coefficients q_j^M. However, to avoid numerical instability for larger M we expand (3.23) by taking partial expansion of the function Φ. After some algebra we get:

$$\sum_{j=1}^{P} q_j^M \left[r_l^{j-1-D_j} B(r_l)^{-j} + \sum_{s=K+1} r_s^{j-1-D_j} B(r_s)^{-j} A(r_l, r_s) \left(\frac{r_l}{r_s}\right)^M \right] = \delta_{1,l} P(1-\rho) \quad (3.24)$$

where

$$A(r_l, r_s) = \left(\frac{K - \Pr_l B'(r_l)/B(r_l)}{K - \Pr_s B'(r_s)/B(r_s)}\right) \left[\sum_k B^{-1}(r_l, k) B(k, r_s)\right] \quad (3.25)$$

with

$$B(k, r_s) = r_s^{D_k - k} B(r_s)^k$$

and $B^{-1}(r_l, k)$ is the inverse of the matrix $B(k, r_l)$.

The form of (3.24) is well suited for numerical calculations. This is due to the fact that the second part in the equations will decrease by

$$\left(\frac{r_l}{r_s}\right)^M.$$

The form of (3.24) is also useful to determine the overall cell loss probability. Inserting $z = 1$ in (3.19) we obtain the mean volume of cells lost in a period;

$$E[V_L] = \sum_{j=1}^{P} \sum_{s=1}^{\infty} s g_j^M(s) :$$

$$E[V_L] = \sum_{j=1}^{P} q_j^M - P(1-\rho) \quad (3.26)$$

From (3.24) taking $l = 1$ we get the overall cell loss probability

$$p_L = \frac{E[V_L]}{P\rho}$$

as:

$$p_L = \frac{-1}{P\rho} \sum_{j=1}^{P} q_j^M \sum_{s=K+1} r_s^{j-1-D_j} B(r_s)^{-j} A(1, r_s) \left(\frac{1}{r_s}\right)^M \quad (3.27)$$

As for the infinite buffer case the calculations can be carried further if the periodic source contains only one single busy/idle period. In addition we also require only one single backward busy-idle period (for the periodic source). When this is the case we take advantage of the fact that (3.24) will be Vandermonde like:

$$\sum_{j=1}^{K} q_j^M \left[\zeta_l^{j-1} + \sum_{s=K+1} \zeta_s^{j-1} A(r_l, r_s) \left(\frac{r_l}{r_s}\right)^M \right] = \delta_{1,l} P(1-\rho) \quad (3.28)$$

with

$$A(r_l,r_s) = \left(\frac{\zeta_l}{\zeta_s}\right)^{-L}\left(\frac{KB(r_l) - Pr_lB'(r_l)}{KB(r_s) - Pr_sB'(r_s)}\right)\left[\frac{\prod_{k=1,k\neq l}^{K}\left(\zeta_s^{-1}-\zeta_k^{-1}\right)}{\prod_{k=1,k\neq l}^{K}\left(\zeta_l^{-1}-\zeta_k^{-1}\right)}\right] \quad (3.29)$$

where $\zeta_i = r_i/B(r_i)$ and L is the displacement in slots between the backward busy period and the busy period (for the periodic source) (see Figure 3).

The generating function for the q_j^M's in this case may be expressed as:

$$\sum_{j=1}^{K} q_j^M x^{j-1} = P(1-\rho)E_1(I+W)^{-1}V(x) \quad (3.30)$$

where E_1 is the K dimensional row vector $E_1 = (1,0,....,0)$, and $W = (W(i,l))$ is the $K \times K$ matrix:

$$W(i,l) = \sum_{s=K+1}\left(\frac{r_l}{r_s}\right)^{M}\left(\frac{\zeta_l}{\zeta_s}\right)^{K-L-1}\left(\frac{KB(r_l)-Pr_lB'(r_l)}{KB(r_s)-Pr_sB'(r_s)}\right)V_i(\zeta_s)V_l(\zeta_s) \quad (3.31)$$

and $V(x)$ is the K dimension column vector $V(x) = (V_l(x))$:

$$V_l(x) = \frac{\prod_{k=1,k\neq l}^{K}(x-\zeta_k)}{\prod_{k=1,k\neq l}^{K}(\zeta_l-\zeta_k)} \quad (3.32)$$

The overall cell loss probability for this special case may be written:

$$p_L = -\frac{(1-\rho)}{\rho}E_1(I+W)^{-1}WE_1^T \quad (3.33)$$

(where E_1^T is the transposed of E_1).

For large M we can expand $(I + W)^{-1}$ in (2.33), and by taking only the first term in the expression for W we get the following simple approximate formula for the overall cell loss probability:

$$p_L \approx \frac{(1-\rho)^2\left(\frac{1}{r_{K+1}}\right)^{M}\left(\frac{1}{\zeta_{K+1}}\right)^{K-L-1}}{\rho\left(\frac{P}{K}\left(r_{K+1}B'(r_{K+1})-B(r_{K+1})\right)\right)}\wp^2 \quad (3.34)$$

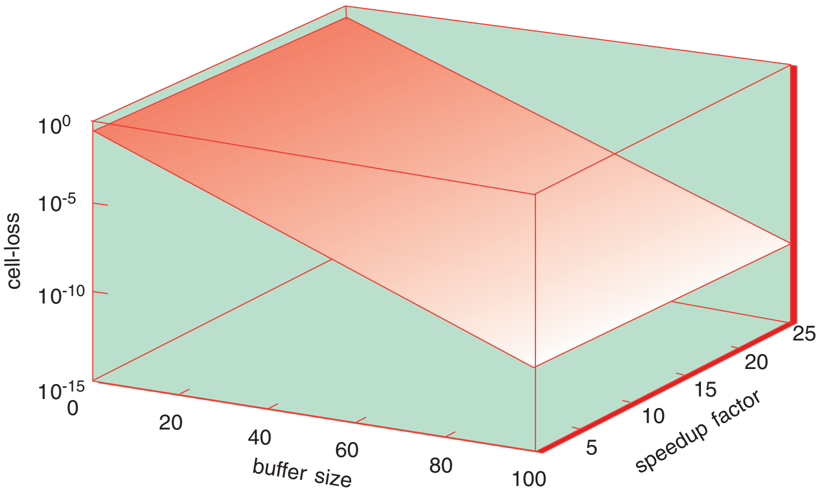


Figure 4 Cell loss probability as a function of buffer size and speed up factor at load 0.9

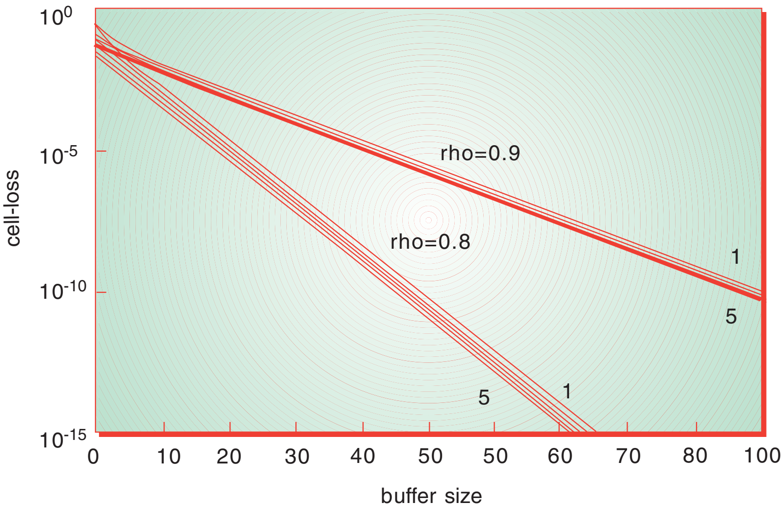


Figure 5 Cell loss probability as a function of buffer size for different speed up factors at load 0.8 and 0.9 (1 speed up = 1, 2 speed up = 5, 3 speed up = 10, 4 speed up = 15, 5 speed up = 20)

where $\wp$ is the product

$$\wp = \frac{\prod_{k=2}^{K}(\zeta_{K+1}-\zeta_k)}{\prod_{k=2}^{K}(1-\zeta_k)} \quad (3.35)$$

For most of the cases of practical interest the formulas (3.34), (3.35) give satisfactory accuracy.

As for the multiserver case the product $\wp$ may be given as an integral. We must, however, substitute for

$$\zeta = \frac{z}{B(z)}$$

in the formulas (2.13) and (2.14) (and also changing n to K and $A(z)$ to $B(z)^P$):

$$\wp = \frac{(1-\rho_B)}{P(1-\rho)}\frac{(\zeta_{K+1})^K}{(\zeta_{K+1}-1)}\exp(I) \quad (3.36)$$

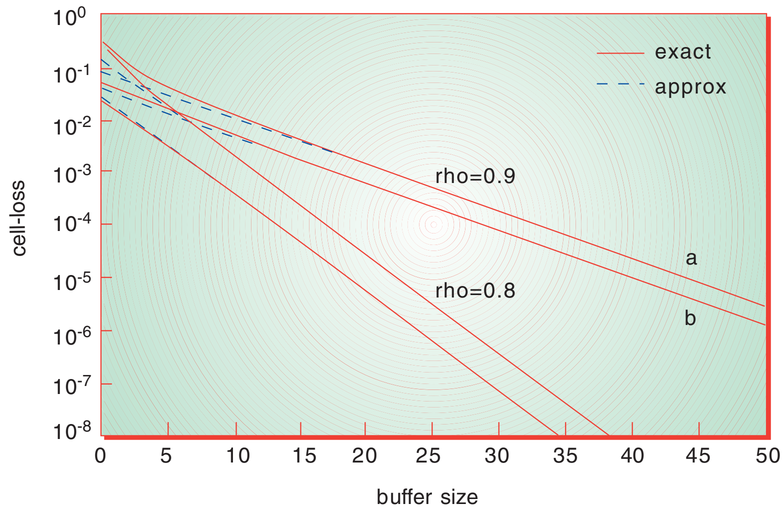


Figure 6 Exact and approximate cell loss probability as function of buffer size for different speed up factor and load 0.8 and 0.9 (a speed up = 1, b speed up = 20)

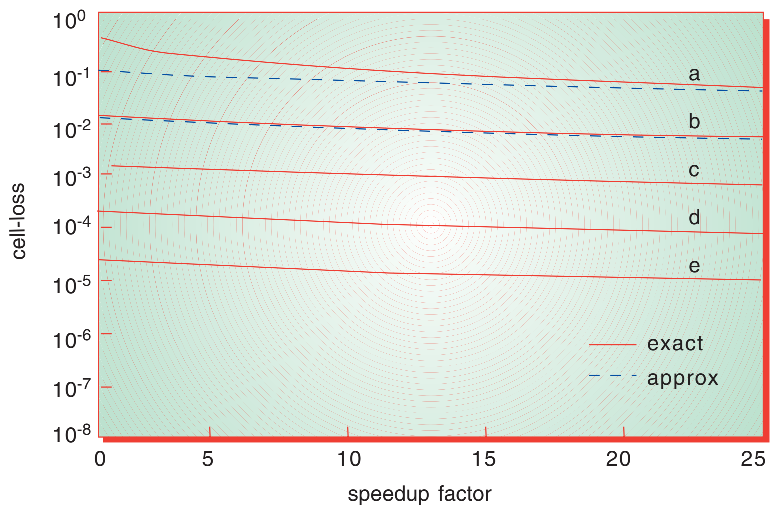


Figure 7 Exact and approximate cell loss probability as a function of the speed up factor for different buffer size and load 0.9 (a buffer = 1, b buffer = 10, c buffer = 20, d buffer =30, e buffer = 40)

where I is the contour-integral

$$I = \frac{1}{2\pi i}\int_{|z|=R} \frac{\zeta'(z)(\zeta_{K+1}-1)}{(\zeta_{K+1}-\zeta(z))(\zeta(z)-1)}$$

$$\log\left(1-\frac{B(z)^P}{z^K}\right)dz \qquad (3.37)$$

where

$$\zeta(z) = \frac{z}{B(z)} \text{ and } \zeta'(z) = \frac{1}{B(z)} - \frac{zB'(z)}{B(z)^2},$$

and R is chosen so that $1 < R < r_{K+1}$.

The main reason for giving the integral formula (3.37) is that for large period P the integral may be approximated by the saddle-point method. The oscillating part in (3.37) is

$$h(z) = \log B(z) - \frac{K}{P}\log z,$$

and we get the following approximate formula for I:

$$I \approx \frac{1}{\sqrt{2\pi P h''(r_M)}}$$

$$\frac{\zeta'(r_M)(\zeta_{K+1}-1)}{(\zeta_{K+1}-\zeta_M)(\zeta_M-1)}$$

$$\log\left(1-\frac{B(r_M)^P}{r_M^K}\right) \qquad (3.38)$$

where and r_M is the root in the interval $(1, r_{K+1})$ of the equation

$$h'(r_M) = \frac{B'(r_M)}{B(r_M)} - \frac{K}{Pr_M} = 0 \qquad (3.39)$$

and

$h''(r_M)$ =

$$\frac{B''(r_M)}{B(r_M)} - \frac{B'(r_M)^2}{B(r_M)^2} + \frac{K}{Pr_M^2} \qquad (3.40)$$

4 Some numerical results

The first part of the numerical results is an investigation of buffer dimensioning of an ATM multiplexer. The results focus mainly on the cell loss probability as function of load, buffer size and speed up factor. The arrival process is taken to be Poisson.

In Figures 4 and 5 we have used the exact formula (2.48) to calculate the cell loss probability as a function of the buffer size and speed up factor. The matrix *W(i,j)* is calculated by (2.44) and the roots are found by an iterative procedure. We see that the effect of the speed up factor is relative weak, the cell loss probability slowly decreases as a function of increasing speed up factor. Apart for buffers of length 20 (or less), the cell loss is nearly log linear as a function of the buffer size.

In Figures 6 and 7 we compare the approximate formula for the cell loss probability with the exact results. From the figures we observe that unless the buffers are small (approx. 20) the approximate formulae for the cell loss gives excellent results. We conclude that for all practical applications the approximations (given by (2.49) and (2.50)) are sufficient for dimensioning of buffers.

In terms of buffers needed, we observe that 50 buffers is sufficient for a link load up to 80% ($\rho = 0.8$) to meet a cell loss demand of 10^{-10}. If the link load is limited to 90% ($\rho = 0.9$) the need for buffer space is doubled (approx. 100). We must however stress that these results are based on the assumption of Poisson arrivals.

The second part of the numerical investigation deals with the effect of superposing a periodic on/off source with a Poisson background stream. The period is set to 100 cell times, and we increase the length of the on period (and thereby increasing the load). In Figures 8 and 9 we have used the exact formula (3.33) to calculate the cell loss probability as a function of the buffer size and load from the background source, while keeping the total load fixed. The matrix $W(i,j)$ is calculated by (3.31) and the roots are found by an iterative procedure.

From Figure 9 we see the effect of increasing the on period (for the periodic source). For small buffers the increase results in worse performance. This is due to the fact that the buffer is not able to absorb all the cells in an on period (and a part of them is lost). However, for larger buffers this is not the case and it is likely that all the cells in an on period are buffered without any losses. (The curves seem to have a common intersection point at approx. 50.) We may say that a multiplexer with small buffers see a periodic on/off source as a bursty source, however for larger buffers this will not be the case.

In Figures 10 and 11 we compare the approximate formula for cell loss probability ((3.34) and (3.35)) with the exact results. The approximation is best when the load from the background stream is high (compared with the load from the periodic source). As the buffer space increases the difference between exact and approximate values decreases rapidly.

5 Conclusions

In this paper we consider two queuing models for dimensioning of buffers and links in an ATM network. The first model is a discrete time multiserver queuing system with batch arrivals. The second model is a single server queue where the multiplexer is fed by the superposition of a periodic stream and a batch arrival process. Both infinite and finite buffers are consider and approximate formulae for the cell loss probability are given. The numerical examples shows that the approximate formulae are well suited for dimensioning purposes.

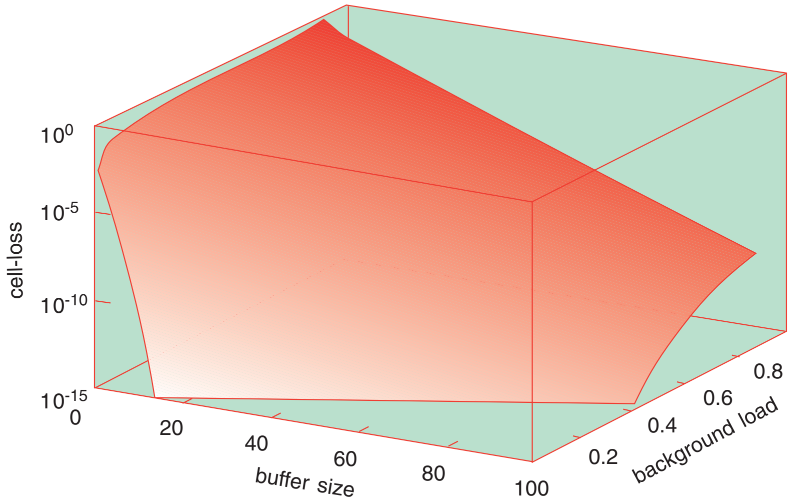


Figure 8 Cell loss probability as a function of buffer size and background load. The period is 100 and total load 0.9

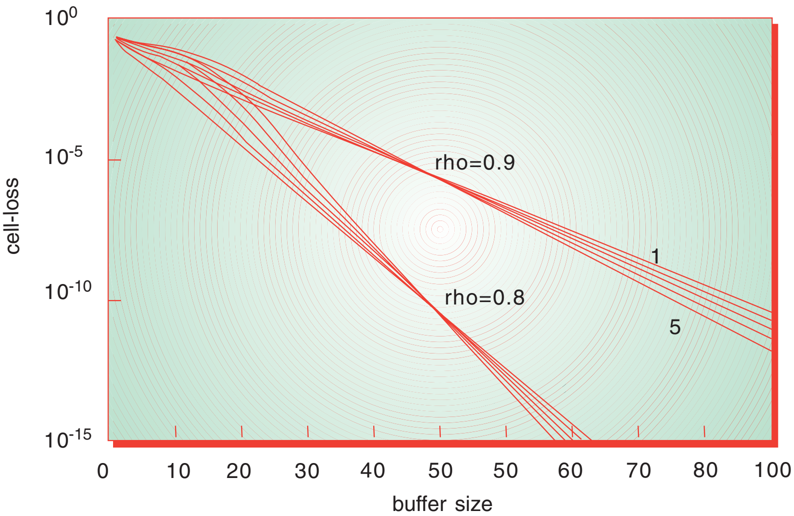


Figure 9 Cell loss probability as a function of buffer size for different loads from the periodic source. The period is 100 and total load 0.8 and 0.9 (1 per. load = 0.05, 2 per. load = 0.10, 3 per. load = 0.15, 4 per. load = 0.20, 5 per. load = 0.25)

References

1 Roberts, J W (ed). *Performance evaluation and design of multiservice networks.* Commission of The European Communities, 1991.

2 Simonian, A, Brown P. Perturbation of a periodic flow in a synchronous server. I: *Proceedings performance 87,* North-Holland, Elsevier, 1987, 89–112.

3 Murata, M et al. Analysis of a discrete-time single-server queue with bursty inputs for traffic control in ATM networks. *IEEE journal on selected areas in communications,* 8, (3), 1990.

4 Bruneel, H et al. An analytical technique for the derivation of the delay performance of ATM switches with multiserver output queues. *International journal of digital and analog communication system,* 5, 1992.

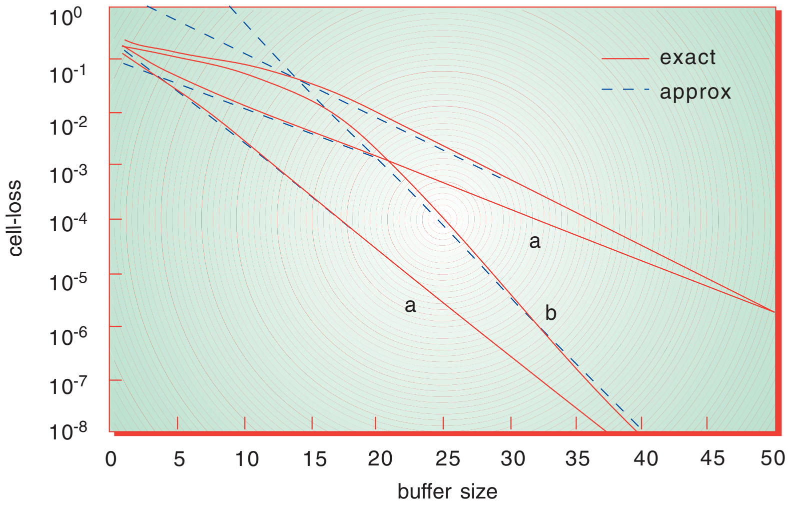


Figure 10 Exact and approximate cell loss probability as a function of buffer size for different loads from the periodic source. The period is 100 and total load 0.8 and 0.9 (a per. load = 0.05, b per. load = 0.25,)

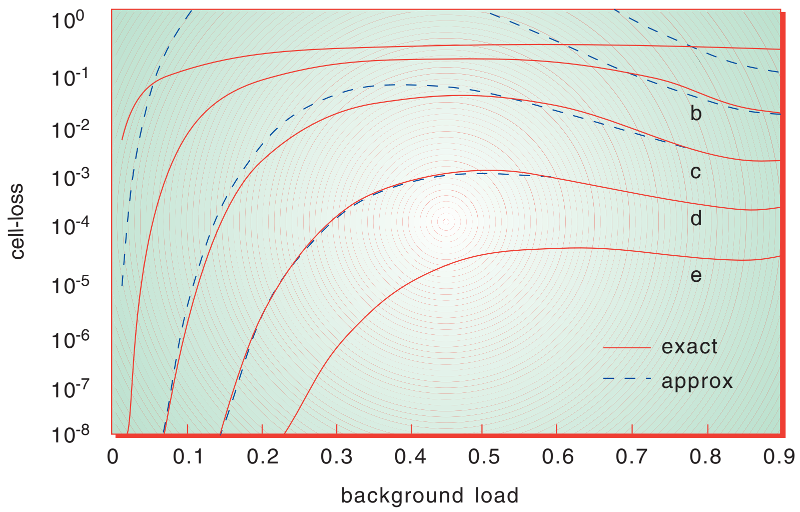


Figure 11 Exact and approximate cell loss probability as function of the background load for different buffer size. The period is 100 and total load 0.9. (a buffer = 1, b buffer =10, c buffer = 20, d buffer =30, e buffer = 40)

5 Saaty, T L. *Elements of queuing theory.* New York, McGraw-Hill, 1961.

6 Wong, R. *Asymptotic approximation of integrals.* Boston, Academic Press, Inc., 1989.

7 Heiss, H. Impact of jitter on peak cell rate policing with a leaky bucket. I: *R1022 workshop on traffic and performance aspects in IBCN,* Aveiro, Portugal, 1992.

Appendix A
Some results for the batch arrival streams considered

The generating functions for Poisson *(M),* Bernoulli *(Geo)* and generalised negative binomial *(Nb)* arrival processes are given by

$$A_M(z) = e^{-A(1-z)} \qquad (A.1)$$

$$A_{Geo}(z) = \left(1 - \frac{A}{N} + \frac{A}{N}z\right)^N \qquad (A.2)$$

$$A_{Nb}(z) = \left(1 + \frac{A}{N} - \frac{A}{N}z\right)^{-N} \qquad (A.3)$$

where we have chosen the parameters so that A is the offered traffic (and N is the extra parameter in the Bernoulli and the generalised negative binomial processes). The variance is given by

$$\sigma_M^2 = A, \sigma_{Geo}^2 = A\left(1 - \frac{A}{N}\right)$$

and

$$\sigma_{Nb}^2 = A\left(1 + \frac{A}{N}\right)$$

and thus we may fit N to a given variance depending on its magnitude. (For the case $\sigma^2 < A$ one must be aware of the fact that the Bernoulli distribution requires N to be an integer.)

By inverting $A(z)^k$ and $A(z)^{-k}$, for the three cases above we get

$$a_{M_i}^k = \frac{(kA)^i}{i!}e^{-kA} \qquad (A.4)$$

$$a_{Geo_i}^k = \binom{Nk}{i}\left(\frac{A}{N}\right)^i\left(1 - \frac{A}{N}\right)^{Nk-i} \qquad (A.5)$$

$$a_{Nb_i}^k = \binom{Nk+i-1}{i}\left(\frac{A}{N}\right)^i\left(1 + \frac{A}{N}\right)^{Nk-i} \qquad (A.6)$$

and

$$a_{m_i}^{-k} = (-1)^i\frac{(kA)^i}{i!}e^{kA} \qquad (A.7)$$

$$a_{Geo_i}^{-k} = (-1)^i\binom{Nk+i-1}{i}\left(\frac{A}{N}\right)^i\left(1 - \frac{A}{N}\right)^{-(i+Nk)} \qquad (A.8)$$

$$a_{Nb_i}^{-k} = (-1)^i\binom{Nk}{i}\left(\frac{A}{N}\right)^i\left(1 + \frac{A}{N}\right)^{Nk-1} \qquad (A.9)$$

Appendix B Evaluation of two logarithmic sums

To derive (2.12) and (2.13) we denote

$$S = \sum_{k=2}^{n} \log(z - r_k)$$

(where we use theprincipal value of the logarithm). We also denote

$$g(\zeta) = 1 - \frac{A(\zeta)}{\zeta^n}.$$

The contour-integral

$$\frac{1}{2\pi i}\int_\Gamma \log(\zeta - z)\frac{g'(\zeta)}{g(\zeta)}d\zeta$$

$$= \sum_{k=1}^{n} \log(z - r_k) - n\log z \qquad \text{(B.1)}$$

where Γ is a contour containing all the roots $r_1 = 1, r_2, \ldots, r_n$ of $g(\zeta)$ and also contains $\zeta = 0$ (which is a pole of multiplicity n for $g(\zeta)$). Whence

$$S = \frac{1}{2\pi i}\int_\Gamma \log(\zeta - z)\frac{g'(\zeta)}{g(\zeta)}d\zeta$$

$$+ n\log z - \log(z - 1) \qquad \text{(B.2)}$$

Depending on the location of z we choose some different contours. First we let C be the circle $|\zeta| = r$ with $1 < r < r_{n+1}$

if $|z| > r$ we choose the circle C as the contour Γ (see Figure 12).

Integrating by parts we get

$$\frac{1}{2\pi i}\int_C \log(\zeta - z)\frac{g'(\zeta)}{g(\zeta)}d\zeta$$

$$= \frac{1}{2\pi i}[\log(\zeta - z)\log g(\zeta)]_C$$

$$-\frac{1}{2\pi i}\int_C \frac{\log g(\zeta)}{\zeta - z}d\zeta \qquad \text{(B.3)}$$

When ζ is moving around C the argument of $\log g(\zeta)$ returns to its initial value. Therefore, $[\log(\zeta - z)\log g(\zeta)]C = 0$.

Collecting the result above gives:

$$S = \log\frac{z^n}{z-1} + \frac{1}{2\pi i}\int_C \frac{\log g(\zeta)}{z - \zeta}d\zeta \qquad \text{(B.4)}$$

If $|z| < r$ we choose the contour $\Gamma = C \cup L_1 \cup L_2 \cup C_\varepsilon$ (see Figure 13).

On C integrating by parts we get (B.3), however, when ζ moves along C from z_0 the argument of $\log(\zeta - z)$ is increased by 2π, while the argument of $\log g(\zeta)$ returns to its starting value, so

$$\frac{1}{2\pi i}[\log(\zeta - z)\log g(\zeta)]_C = \log g(z_o)$$

On $L_1 \cup L_2$ we have

$$\frac{1}{2\pi i_{L_1}}\int_{\cup L_2} (\ )d\zeta =$$

$$\frac{1}{2\pi i}\int_{z+\varepsilon}^{z_0} \log(\zeta - z)\frac{g'(\zeta)}{g(\zeta)}d\zeta +$$

$$\frac{1}{2\pi i}\int_{z_0}^{z+\varepsilon} \log(\zeta - z)\frac{g'(\zeta)}{g(\zeta)}d\zeta$$

On L_2 the argument of $\log(\zeta - z)$ has increased by 2π so that

$$\frac{1}{2\pi i_{L_1}}\int_{\cup L_2} (\ )d\zeta =$$

$$-\frac{2\pi i}{2\pi i}\int_{z+\varepsilon}^{z_0} \frac{g'(\zeta)}{g(\zeta)}d\zeta =$$

$$-\log g(z_0) + \log g(z + \varepsilon) \qquad \text{(B.5)}$$

On C_ε we have $\zeta = z + \varepsilon e^{i\theta}$ where θ moves from 2π to 0. (We also assume that $z \neq r_i$, $i = 1, \ldots, n$.) When $\varepsilon \to 0$ we have

$$\frac{1}{2\pi i}\int_{C_\varepsilon} \log(\zeta - z)\frac{g'(\zeta)}{g(\zeta)}d\zeta \sim$$

$$\frac{g'(\zeta)}{g(\zeta)}\frac{1}{2\pi}\int_{2\pi}^{0} \log(\varepsilon e^{i\theta})(\varepsilon e^{i\theta})d\theta \to 0 \qquad \text{(B.6)}$$

Collecting the results above when $\varepsilon \to 0$ we get

$$S = \log\frac{z^n g(z)}{z-1} + \frac{1}{2\pi i}\int_C \frac{\log g(\zeta)}{z - \zeta}d\zeta \qquad \text{(B.7)}$$

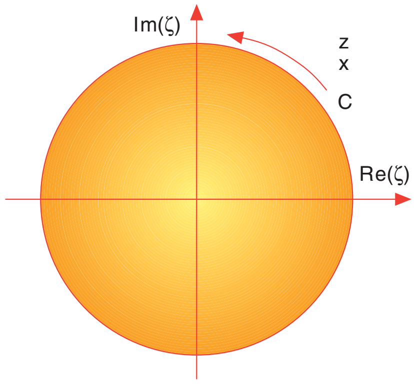


Figure 12 The contour Γ when $|z| > r$

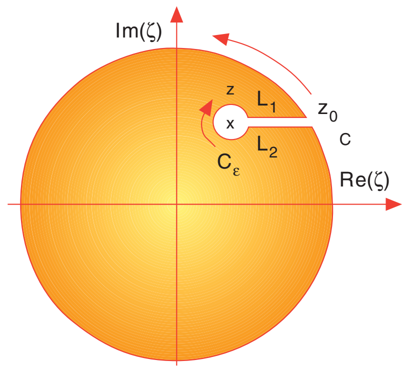


Figure 13 The contour Γ when $|z| < r$

Notes on a theorem of L. Takács on single server queues with feedback

BY ELIOT J JENSEN

Abstract

L. Takács published in 1963 a paper introducing a queuing model with feedbacks. Such models may be used to calculate the performance of certain telecommunication processor systems. This note proposes alternative formulations of some of the results in the paper. They may be convenient when calculating response times for strings of processing tasks representing e.g. call handling functions. The impact of task scheduling on the range of response times has been indicated.

Background

L. Takács [1] examines the system sojourn times of tasks offered to a single server, with an infinite queue in front of it. Tasks arrive to the system either from outside, according to a Poisson process with intensity λ, or as feedbacks. A feedback may be generated at the termination of the service of a task, and immediately put at the tail of the queue. The probability for this is p and for no feedback $q = 1 - p$. Tasks are assumed to have independent but identically distributed service times. This distribution is assumed to have the Laplace-Stieltje transform $\Psi(s)$. Let its mean value be α. ***Theorem 3*** of the Takács paper considers the stationary distribution of the total sojourn time θ_n (n = 1, 2, ...) for a random task and its feedbacks, and reads, cit:

> *If $\lambda\alpha < q$, then θ_n has a unique stationary distribution $P\{\theta_n \leq x\}$, which is given by the following Laplace-Stieltje transform*
>
> $$\Phi(s) = q\sum_{k=1}^{\infty} p^{k-1} U_k(s,1) \ (\mathcal{R}(s) \geq 0) \quad (1)$$
>
> *where*
>
> $$U_1(s,z) = P_0\Psi(s + \lambda(1 - z)) + U(s + \lambda(1 - z), (q + pz)\Psi(s + \lambda(1 - z))) \quad (2)$$
>
> *for $\mathcal{R}(s) \geq 0$ and $|z| \leq 1$, $P_0 = 1 - \lambda\alpha/q$, $U(s,z)$ is defined by (20) of [1], and*
>
> $$U_{k+1}(s,z) = \Psi(s + \lambda(1 - z)) U_k(s,(q + pz)\Psi(s + \lambda(1 - z))) \quad (3)$$
>
> *for k = 1, 2, ...*

The formula $U(s,z)$ (eq. 20 of [1]) is defined as the combined Laplace-Stieltje transform and z-transform of a distribution $P_j(t)$, giving the probability of, in the stationary state (assuming $\lambda\alpha < q$), to encounter exactly j (> 0) tasks in the system, where the task in the server has remaining service time shorter or equal to t. It reads:

$$U(s,z) = \left(1 - \frac{\lambda\alpha}{q}\right) \frac{\lambda z(1-z)(\Psi(s) - \Psi(\lambda(1-z)))}{(z - (q+pz)\Psi(\lambda(1-z)))(s - \lambda(1-z))} \quad (4)$$

We also note that $U_k(s,z)$ is the Laplace-Stieltje transform and z-transform of the combined distribution of the total sojourn time of a task and its feedbacks, and the number of tasks left behind when the last of its feedbacks leaves the server, provided the number of feedbacks is exactly $k - 1$ for that particular task.

From ***theorem 3*** is obtained

$$\Phi(s,z) = q\sum_{k=1}^{\infty} p^{k-1} U_k(s,z) = qU_1(s,z) + p\Psi(s + \lambda(1-z)) \Phi(s, (q+pz)\Psi(s+\lambda(1-z))) \quad (5)$$

Then the moments

$$E\{\theta_n^r\} = (-1)^r \left(\frac{\partial^r}{\partial s^r}\Phi(s,z)\right)_{\binom{s=0}{z=1}}$$

for the stationary sojourn time process may be found by solving a set of $r + 1$ linear equations for the determination of

$$\Phi_{ij} = \left(\frac{\partial^{i+j}\Phi(s,z)}{\partial s^i \partial z^j}\right)_{\binom{s=0}{z=1}} \quad (6)$$

Takács has given explicit formulae for $E\{\theta_n{}^r\}$ for r = 1 and 2.

An alternative formulation

In the recurrence formula for $U_k(s,z)$ in ***theorem 3***, make the substitution

$$\xi_1(s,z) = (q + pz)\Psi(s + \lambda(1 - z)) \quad (7)$$

and introduce the recurrence

$$\xi_k(s,z) = (q + p\xi_{k-1}(s,z)) \Psi(s + \lambda(1 - \xi_{k-1}(s,z))) \quad (8)$$

starting with $\xi_0(s,z) = z$. Then, by rearrangement we obtain

$$U_{k+1}(s,z) = \left(\prod_{\ell=0}^{k-1} \Psi(s + \lambda(1 - \xi_\ell(s,z)))\right) U_1(s, \xi_k(s,z)) \quad (9)$$
$(k = 0, 1, 2, ...)$

It is easily seen that $\xi_\ell(s,z)$ is the Laplace-Stieltje transform and z-transform of the combined distribution of system sojourn time and the number of tasks left in the system for a task which needs to enter the server ℓ times, but always with zero service time, the task joining the system finding only one (other) task there, which is just about to enter service. It follows that

$$\lim_{\ell \to \infty} \xi_\ell(s,z) = \bar{B}(s)$$

for $|z| \leq 1$ and $\lambda\alpha < q$.

$\bar{B}(s)$ is the Laplace-Stieltje transform of the busy period distribution.

For convenience the formula for $U_k(s,z)$ may be slightly rewritten, using the z-transform and the Laplace-Stieltje transform of the number of tasks in the system and the remaining service time of the task in the server for the stationary process at a random time. If the system is empty, the remaining service time is zero. Based on theorem 1 of [1] we may put

$$P(s,z) = \left(1 - \frac{\lambda\alpha}{q}\right) \left(1 + z\frac{\lambda(1-z)}{s - \lambda(1-z)} \cdot \frac{\Psi(s) - \Psi(\lambda(1-z))}{z - (q+pz)\Psi(\lambda(1-z))}\right) \quad (10)$$

A task (which we may call a tagged task) arriving at a random instant to the system will see this distribution. When the remaining service time finishes, a feedback may, or may not be generated. At the moment the server is about to start processing the next task, the tagged task will see a number i of tasks ahead of it in the queue and a number j of tasks behind, and at that time it will have waited a certain time t (identical to the remaining service time of the task initially in the server). The Laplace-Stieltje and z-transform of the corresponding distribution is

$$Q(s,y,z) = \left(1 - \frac{\lambda\alpha}{q}\right)\left(1 + (q+pz) \frac{\lambda(1-y)}{s + \lambda(1-z) - \lambda(1-y)} \cdot \frac{\Psi(s+\lambda(1-z)) - \Psi(\lambda - y))}{y - (q+py)\Psi(\lambda(1-y))}\right) \quad (11)$$

At the moment our test task enters the server it will have waited a time t and there will be j other tasks in the system. The corresponding distribution becomes

$$V_1(s,z) \\ = Q(s,(q+pz)\Psi(s+\lambda(1-z)),z) \\ = Q(s,\xi_1(s,z),\xi_0(s,z)) \quad (12)$$

Assume that our tagged task is a zero-service-time task and that it passes the server k times. When it passes for the k'th time it will have been in the system a time t and there will be j other tasks in the system. The corresponding distribution will have the transform

$$V_k(s,z) = V_1(s,\xi_{k-1}(s,z)) \quad (k = 1, 2, ...) (13)$$

We are now in the position to rewrite $U_k(s,z)$ as

$$U_k(s,z) = \left(\prod_{\ell=0}^{k-1} \Psi(s+\lambda(1-\xi_\ell(s,z)))\right) V_k(s,z) \quad (14)$$

Expressions for the moments of the sojourn time distribution corresponding to $U_k(s,1)$ may be found in terms of moments of the sojourn times of zero service time tasks, i.e.

$$\left(\frac{d^r}{ds^r}\xi_\ell(s,1)\right)_{s=0} \quad (15)$$

by taking the logarithm of equation (14) and differentiating successively. Using this procedure also makes it possible to develop the formulas for the first two moments (which still have a relatively comprehensible structure) of the distribution with transform

$$\Phi(s) = q\sum_{k=1}^{\infty} p^{k-1} U_k(s,1) \\ (\mathcal{R}(s) \geq 0) \quad (16)$$

as dealt with in the paper of Takács.

Application to a deterministic sequence of tasks

Feedback queuing has been used in processor performance models, cfr. e.g. [2], [3] and [4].

A processor in a telecommunication system normally performs an extensive set of different functions, each initiating a deterministic sequence of individual processing jobs. The processing times may be deterministic or have specific distributions.

Such a processor may provide different performance, in terms of e.g. response times, for each of the various functions. For each specific function the performance will depend on

1 the whole set of functions offered and their arrival intensities, i.e. on the workload characteristics. This may be modeled by means of a single server queuing process with feedback, as in the paper of Takács, but possibly with a slightly more general distribution f_i for the number of feedbacks which may be generated as a result of the processing of a job. The external arrival intensity and the service time distribution $\Psi(t)$ will depend on the mix of functions and their frequencies. One may call this queuing process the background process,

2 the processing requirement of the specific function and the way it is handled by the operating system in terms of a sequence of processor jobs. This may be modeled as a sequence of tagged jobs with service times according to different distributions. Some, if not most, of the processing times will be deterministic.

To derive the response time of a specific function comprising a sequence of k tagged jobs one has to make changes corresponding to the introduction of different service time distributions in the formula

$$U_k(s,z) = \left(\prod_{\ell=1}^{k} \Psi(s+\lambda(1-\xi_{k-\ell}(s,z)))\right) V_k(s,z) \quad (17)$$

which is just a rearranged formula (14). The corresponding response time / number of left jobs distribution will have the transform

$$W_k(s,z) = \left(\prod_{\ell=1}^{k} \Psi_\ell(s+\lambda(1-\xi_{k-\ell}(s,z)))\right) V_k(s,z) \\ (k = 1, 2, ...) \quad (18)$$

where Ψ_1 is the Laplace-Stieltje transform of the processing time distribution for tagged job No. 1

If all the tagged jobs have deterministic processing times this formula may be written as

$$W_k\left(s;\{t_\ell\}_{\ell=1}^{k};z\right) = \\ e^{-\sum_{\ell=1}^{k}(s+\lambda(1-\xi_{k-\ell}(s,z)))t_\ell} V_k(s,z) \quad (19) \\ (k = 1, 2, ...)$$

Note that the definition of $\xi_\ell(s,z)$ now is slightly different from the previous chapter, as we will have

$$\xi_\ell(s,z) = f(\xi_{\ell-1}(s,z)) \\ \Psi(s+\lambda(1-\xi_{\ell-1}(s,z))) \\ (\ell = 1, 2, ...) \quad (20)$$

and $\xi_0(s,z) = z$. $f(z)$ is the z-transform of the feedback distribution f_i.

Stationarity of the process is assured whenever the load

$$\rho = \frac{\lambda\alpha}{1-f'(1)} \quad (21)$$

of the server is less than 1.

Also the definition of $V_k(s,z)$ has to be slightly changed. Eq. (13) is still valid, but now starts with:

$$V_1(s,z) = \\ (1-\rho)\left(1+\frac{\lambda(1-\xi_1(s,z))}{s+\lambda(\xi_1(s,z)-z)} \cdot \frac{\xi_1(s,z)-f(z)\Psi(\lambda(1-\xi_1(s,z)))}{\xi_1(s,z)-f(\xi_1(s,z))\Psi(\lambda(1-\xi_1(s,z)))}\right) \quad (22)$$

Of course, the $U_k(s,z)$ discussed previously may be obtained by assuming that all tagged jobs have processing times with the same distribution as the other jobs, and that the feedback distribution has z-transform $f(z) = q + pz$.

The form of the response time distribution is conceptually convenient inasmuch as it illustrates the impact of the partition of work of each particular function on its system response time.

The mean and variance of the response times are easily calculated.

$$\bar{t}_{w_k} = \bar{t}_{v_k} + \sum_{\ell=1}^{k}\left(1+\lambda\xi_{k-\ell}^{(1)}\right)\cdot\bar{t}_{\Psi_\ell} \quad (23)$$

$$\sigma_{w_k}^2 = \sigma_{v_k}^2 + \sum_{\ell=1}^{k}\left(\left(1+\lambda\xi_{k-\ell}^{(1)}\right)^2 \\ \sigma_{\Psi_\ell}^2 + \lambda\xi_{k-\ell}^{(2)}\bar{t}_{\Psi_\ell}\right) \quad (24)$$

where

$$\xi_k^{(1)} = -\left(\frac{d}{ds}\xi_k(s,1)\right)_{s=0}$$

$$= h^{(1)}\frac{1-\left(c_1+\lambda h^{(1)}\right)^k}{1-c_1-\lambda h^{(1)}} \qquad (25)$$

and

$$\xi_k^{(2)} = \left(\frac{d^2}{ds^2}\xi_k(s,1)\right)_{s=0}$$

$$= \frac{c_3+2c_1\lambda h^{(1)}+\lambda^2 h^{(2)}}{(1-c_1-\lambda h^{(1)})^2}\cdot$$

$$h^{(1)}\left\{\left(1+\left(c_1+\lambda h^{(1)}\right)^{k-1}\right)\xi_k^{(1)}\right.$$

$$\left.-2kh^{(1)}\left(c_1+\lambda h^{(1)}\right)^{k-1}\right\} \qquad (26)$$

The parameters c_k are the factorial moments of the feedback distribution f_j, and $h^{(1)}$ the moments of the background processing time distribution $\Psi(t)$ with respect to $t = 0$.

The average sojourn time for a zero-service-time task passing the server k times is

$$\bar{t}_{v_k} = (1-\rho)\left(h^{(1)}a_k + \frac{\lambda b_k}{1-c_1-\lambda h^{(1)}}\right) \qquad (27)$$

where

$$a_k = \frac{1}{2}\lambda\frac{c_2+2c_1\lambda h^{(1)}+\lambda^2 h^{(2)}}{\left(1-c_1-\lambda h^{(1)}\right)^2}\xi k_k^{(1)} \qquad (28)$$

and

$$b_k = c_1 h^{(1)}\xi_{k-1}^{(1)}$$

$$+\frac{1}{2}h^{(2)}\left(1+\lambda\xi_{k-1}^{(1)}+\lambda\xi_k^{(1)}\right) \qquad (29)$$

The variance of the sojourn time may be found as

$$\sigma_{v_k}^2 = (1-\rho)$$

$$\left\{h^{(1)}p_k + 2a_k b_k + \frac{\lambda q_k}{1-c_1-\lambda h^{(1)}}\right\}$$

$$-\left(\bar{t}_{v_k}\right)^2 \qquad (30)$$

where

$$p_k = \frac{\lambda}{\left(1-c_1-\lambda h^{(1)}\right)^2}$$

$$\left\{\frac{c_2+2c_1\lambda h^{(1)}+\lambda^2 h^{(2)}}{2}\xi_k^{(2)}\right.$$

$$+\left(\frac{\left(c_2+2c_1\lambda h^{(1)}+\lambda^2 h^{(2)}\right)^2}{2\left(1-c_1-\lambda h^{(1)}\right)}\right.$$

$$\left.+\frac{c_3+3c_2\lambda h^{(1)}+3c_1\lambda^2 h^{(2)}+\lambda^3 h^{(3)}}{3}\right)$$

$$\left.\left(\xi_k^{(1)}\right)^2\right\} \qquad (31)$$

and

$$q_k = \frac{1}{3}h^{(3)}\left\{\left(1+\lambda\xi_{k-1}^{(1)}\right)^2\right.$$

$$\left.+\lambda\xi_k^{(1)}\left(1+\lambda\xi_{k-1}^{(1)}\right)+\lambda^2\left(\xi_k^{(1)}\right)^2\right\}$$

$$+c_2 h^{(1)}\left(\xi_{k-1}^{(1)}\right)^2$$

$$+c_1 h^{(2)}\xi_{k-1}^{(1)}\left(1+\lambda\xi_{k-1}^{(1)}+\lambda\xi_k^{(1)}\right)$$

$$+\left(c_1 h^{(1)}+\frac{1}{2}\lambda h^{(2)}\right)\xi_{k-1}^{(2)}$$

$$+\frac{1}{2}\lambda h^{(2)}\xi_k^{(2)} \qquad (32)$$

The parameters given by the formulae (23) – (32) all form monotonously increasing sequences (with respect to k). This leads us to conclude that a function with a total processing time distribution with mean value $\bar{t}_p$ and variance σ_p^2 will have a response time distribution $W(t)$ with parameters $\bar{t}_w$ and σ_w^2 fulfilling:

$$\bar{t}_{w_1} \le \bar{t}_w < \bar{t}_{w_\infty} \text{ and } \sigma_{w_1}^2 \le \sigma_w^2 < \sigma_{w_\infty} \qquad (33)$$

where the left sides of the inequalities correspond to the case where all the function is processed as one single $(\bar{t}_p, \sigma_p^2)$ job, and the right side to the case where the processing is portioned into an infinite sequence of jobs, the first comprising the entire processing time $(\bar{t}_p, \sigma_p^2)$, the others constituting zero-processing-time passes. We note that, depending on the actual design of the function and the influence of the operating system, the response time parameters may approach the limits arbitrarily close.

In order to estimate the gap between upper and lower limits, consider that

$$\xi_\infty^{(1)} = \lim_{k\to\infty}\xi_k^{(1)} = \frac{h^{(1)}}{1-c_1-\lambda h^{(1)}}$$

$$= \frac{\xi_1^{(1)}}{(1-c_1)(1-\rho)} \qquad (34)$$

and

$$\xi_\infty^{(2)} = \lim_{k\to\infty}\xi_k^{(2)}$$

$$= \frac{\left(c_2+2c_1(1-c_1)\right)\left(h^{(1)}\right)^2+(1-c_1)^2 h^{(3)}}{\left(1-c_1-\lambda h^{(1)}\right)^3}$$

$$= \frac{c_2+2c_1(1-c_1)}{(1-c_1)^3(1-\rho)^3}\left(h^{(1)}\right)^2$$

$$+\frac{\xi_1^{(2)}}{(1-c_1)(1-\rho)^3} \qquad (35)$$

It may be shown that for $\bar{t}_w$ the upper limit is approximately $(1-\rho)^{-1}$ times the lower limit, while for σ_w^2 the corresponding factor is approximately $(1-\rho)^{-2}$ and there will be implementations approaching the limits arbitrarily close. Normally, a better accuracy will be required. This may be obtained for a specific function by identifying the required sequence of jobs in a processor and applying equations (23) and (24).

References

1 Takács, L. A single-server queue with feedback. *Bell system technical journal,* March 1963.

2 Rapp, D. Some queuing models for AXE-10. I: *Ninth international teletraffic congress, ITC 9,* Torremolinos, 1979.

3 Hedberg, I. MM/1-system : väntetidsfördelningen för en kund som efter avslutad betjäning omedelbart ställer sig sist i kön tills han betjänats sammanlagt K gånger, K ≤ 2. I: *Third Nordic teletraffic seminar, NTS 3,* Stockholm, 1980.

4 Jensen, E. *Response times of a front end / host computer system operating under a certain time sharing algorithm.* Kjeller, Norwegian Telecommunications Administration Research Establishment (now Telenor Research & Development), 1982. (Report No. 29/82.)

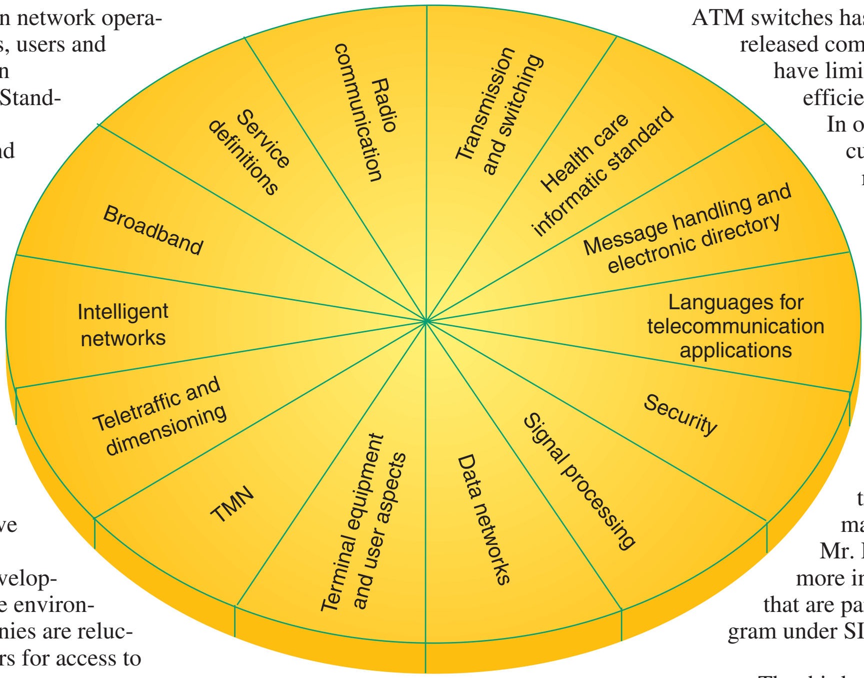
Status

International research and
standardization activities
in telecommunication

Editor: Endre Skolt

Introduction

BY ENDRE SKOLT



On behalf of European network operators, service providers, users and industry, the European Telecommunications Standards Institute (ETSI) work out standards and recommendations as a basis for implementation of telecom equipment in Europe. Today, ETSI have members from 30 countries, and the ETSI results have great impact on the development of telecommunications infrastructure. In recent years many have criticised the lack of speed in standards development. In a competitive environment, telecom companies are reluctant to wait many years for access to standards. In the first paper provided by Mr. *Trond Ulseth*, we take a closer look at the different types of documents and results emerging from ETSI, different types of approval procedures and the weighted voting used to obtain agreements. An ETSI standard will be approved when the percentage of positive votes is at least 71 % of the votes cast.

In the second paper Mr. *Harald Pettersen* presents some ongoing ATM (Asynchronous Transfer Mode) traffic activities. The first generation of ATM switches has already been released commercially, but they have limitations with regard to efficiency and capabilities. In order to improve customer requirements, robustness against network overload, and high network utilisation, several activities have been set up world wide to investigate these issues. In RACE a Special Interest Group (SIG) on traffic control and ATM performance started last year. Mr. Pettersen presents more in detail three projects that are part of the study program under SIG.

The third paper is a brief presentation of a security related study called “The Public Network Operator (PNO) Cipher project” which started February 1 of this year. The objective of the study is to develop a cryptographic algorithm for confidentiality and integrity protection of network management data. Mr. *Øyvind Eilertsen* presents the project background, usage of the algorithm, functional requirements and the “PNO cipher project” work plan.

Table 1 List of contributions to the Status section

Issue No.	Study area	Editor
4.93	Service definitions	Ingvill H Foss
4.93	Radio communications	Ole Dag Svebak
4.93	Transmission and switching	Bernt Haram
4.93	Intelligent Networks	Endre Skolt
4.93	Broadband	Inge Svinnset
1.94	Terminal equipment and user aspects	Trond Ulseth
1.94	Signal processing	Gisle Bjøntegaard
1.94,	Telecommunications Management Network (TMN)	Ståle Wolland
1.94	Teletraffic and dimensioning	Harald Pettersen
1.94	Data networks	Berit Svendsen
2.94	Languages for telecommunication applications	Arve Meisingset
2.94	Message Handling and Electronic Directory	Geir Thorud
2.94	Security	Pål Spilling
3.94	UPT – Service concept and standardisation and the Norwegian pilot service	Frank Bruarøy, Kjell Randsted
3.94	Status report from ITU-T, SG 1	Elisabeth Tangvik
3.94	Future mobile communications	Ole Dag Svebak
3.94	The CIO project	Gabriela Grolms
4.94	Eurescom work on ATM network studies and the European ATM pilot network	Inge Svinnset
4.94	Termina equipment and user aspects	Trond Ulseth
4.94	Telecommunications Management Network	Ståle Wolland

Documents types that are prepared by ETSI

BY TROND ULSETH

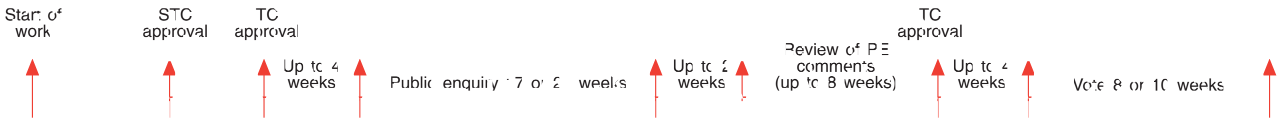


Figure 1

Rules of Procedures and Working Procedures have been prepared for the work of ETSI. These procedures describe the document types that ETSI produces, formal rules for the presentation of the content of these documents, and rules for the approval of the documents.

The document types ETSI produce are

- European Telecommunication Standards (ETS)
- Interim European Telecommunication Standards (I-ETS)
- Technical Bases for Regulation (TBR)
- ETSI Technical Reports (ETRs)
- Technical Committee Reference Technical Reports (TCR-TR)
- Technical Committee Technical Reports (TC-TR).

Standards

Both ETSs and I-ETSs are standards. The approval procedures for ETSs and I-ETSs are identical.

The life of an I-ETS is limited to three years after which it can be

- converted into a European Telecommunication Standard (ETS)
- have its life extended for a further two years
- be replaced by a new version
- be withdrawn.

A standard is given the I-ETS status either because it is regarded as a provisional solution ahead of a more advanced standard, or because it is immature and requires a “trial period”.

Where ETSI has decided to start work on a standard, there is an obligation on all ETSI countries and individuals not to undertake activities which could prejudice the preparation of the ETS defined by the scope statement of the work item. It should be noted that such activities may include not only the preparation of standards, but also that of new or revised technical specifications to support national regulations. These rules, which often are referred to as “Standstill” do not apply to work where the intention is to prepare an I-ETS.

There are three possible approval procedures for standards:

- Normal Approval Procedure (NAP)
- Unified Approval Procedure (UAP)

Maintenance of (Interim) Standards

Whilst every care is being taken in the preparation and publication of ETSI (Interim) Standards, there is always a need for maintenance of approved (Interim) Standards. The maintenance may result in either

a) an Amendment to the current Edition, or
b) a new Edition.

A decision to modify an adopted standard follows the ETSI rules for creating a new work item. At the time of creating a new work item it is not necessary to decide whether the result will be published as an Amendment document or a new Edition.

In general, the following rules apply:

- Amendments shall be avoided whenever possible. They may only be issued in cases of urgently needed improvements of the (Interim) Standard.
- No more than three amendments shall be issued to a single Edition of an (Interim) Standard.
- A new Edition shall be published wherever the size of the text of the amendment approaches 10 % to 15 % of the original text.

Where a new edition is published details about earlier editions shall be contained in the foreword of the document. The number of the edition shall be indicated on the title page of the (Interim) Standard.

All amendments shall have a title page similar to that of the (I-)ETS.

The approval procedure of Amendments or new Editions is normally the UAP. In cases where significant changes to the (I-)ETS are being proposed, the NAP should be used.

- Accelerated Unified Approval Procedure (AUAP).

The Normal Approval Procedure is the normal procedure for approval of standards. It is a two-step procedure with separate Public Enquiry (PE) and Voting phases as described in Figure 1.

The work is normally carried out by a Sub-Technical Committee (STC) or a Project Team (PT). When approved by the Technical Committee (TC) responsible for the work item, the draft is sent to the National Standards Organisations (NSOs) of the ETSI member countries for comments (Public Enquiry). The PE period is usually 17 but is extended to 21 weeks if the PE period includes the summer holiday period.

The review of the PE comments is normally carried by the STC responsible for the draft standard. The PE review takes place at an ordinary STC meeting or at a PE resolution meeting. After TC approval of the modified document, it is sent to the NSOs for vote. The draft is approved if 71 % of the weighted votes are positive.

The approval at STC or TC level is made at meetings or by correspondence. If there are problems reaching a consensus, indicative voting might be necessary. Only the participating Full Members of the STC/TC can vote, and there is one vote for each member. It is for the chairman of the STC/TC to evaluate the results of the indicative voting.

The Unified Approval Procedure (UAP) and the Accelerated Unified Approval Procedure (AUAP) are both one step approval procedures where Public Enquiry and Vote are combined.

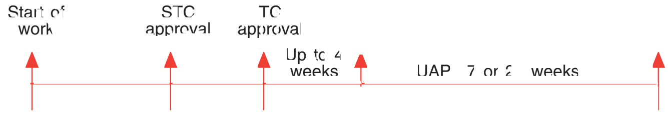


Figure 2

The UAP period is the same as for PE, 17 or 21 weeks. The UAP procedure is usually used for amendments, corrections or revisions of approved standards. Normally, UAP should not be used for more than 20 % of the total cases. The UAP procedure is described in Figure 2.

The AUAP procedure is used only when the Standard is urgently needed to meet the market needs. Careful planning is required, and the NSOs shall be notified before the consensus approval of the STC. Normally, AUAP should not be used for more than 10 % of the total cases.

Technical Bases for Regulation

Technical Bases for Regulation (TBR) is a document that will be used for regulatory purposes in Europe. Work on a TBR will be based on a mandate from CEC and EFTA, and a scope statement defined by the regulatory bodies. A TBR is the technical part of a Common Technical Regulation (CTR) as defined in Council Directive 91/263, and is limited to essential requirements as defined in the directive.

ETSI Weighted National Voting

Draft (I-)ETSs are approved by the ETSI weighted national voting procedure. The vote is usually taken by correspondence, but can be taken at a meeting announced thirty days beforehand.

The draft (I-)ETS is approved when the percentage of the positive votes is at least 71 % of the votes cast. In the events that a draft (I-)ETS is not approved, the national votes of the EC countries shall be counted according to article 148 of the Treaty of Rome. If the vote is positive the standard is then adopted, and thus becomes an (I-)ETS for the European Community and other countries which have voted in its favour.

The allocations of weighting to ETSI member countries are as given in the table. It should be noted that for the time being there are two open issues,

- to reduce the weighting of Austria and Sweden to 4 to be in line with the weighting of the European Community
- to allocate weights for Russia.

Country	Weight	Country	Weight
France	10	Czech Republic	3
Germany	10	Denmark	3
Italy	10	Hungary	3
United Kingdom	10	Finland	3
Spain	8	Ireland	3
Austria	5	Norway	3
Belgium	5	Romania	3
Greece	5	Croatia	2
The Netherlands	5	Cyprus	2
Poland	5	Iceland	2
Portugal	5	Luxembourg	2
Sweden	5	Malta	2
Switzerland	5	Slovak Republic	2
Turkey	5	Slovenia	2
Bulgaria	3	Russia	tba

The rules for preparation and approval of a TBR are the same as for ETSI Standards. For new TBRs the Normal Approval Procedure shall be used. For corrections or amendments the UAP might be used.

In parallel to the ETSI PE the TBR shall be submitted on a GATT notification procedure.

ETSI Technical Report

ETSI Technical Reports (ETR) are prepared to provide

- guidance and supporting information on a Standard or group of Standards and their application
- ETSI recommendations on a subject or subject area which are not of such a nature that they are suitable for an (I-)ETS
- an ETSI common position on an issue which is also being discussed in other bodies
- various types of status reports.

An ETR shall contain a scope statement clearly indicating the nature and status of its contents.

As for standards, ETRs are prepared by an STC or a PT. In cases where there is consensus within a TC the final approval of an ETR takes place at the TC level. However, there is a period of two weeks after the TC approval where objections can be made by any ETSI member. If a consensus cannot be reached, the ETR is passed to ETSI Technical Assembly for further action, e.g. vote at a TA meeting.

ETRs are available to the public.

Technical Committee Reference Technical Report

A Technical Committee Reference Technical Report (TCR-TR) is a document for use within ETSI. It is not available outside ETSI. Requirements contained in a TCR-TR are binding on all ETSI TCs and STCs concerned. TCR-TR may define reference models, terminology, structure of Standards, programme for ETSI projects, etc.

Technical Committee Technical Report

A Technical Committee Technical Report (TC-TR) is a document for use within ETSI TC. It is not available outside ETSI. It documents work within the TC which is of internal interest, but not, in general, of interest to other ETSI TCs. A TC-TR may contain requirements that are binding on the originating TC and its STCs.

ATM traffic activities in some RACE projects

BY HARALD PETTERSEN

Services in an ATM network may be based on different ATM Layer Bearer Capabilities. An ATM layer bearer capability specifies a combination of Quality of Service (QoS) commitments and ATM traffic parameters that are suitable for some applications and that allow for a specific multiplexing scheme at the ATM layer [1]. Several ATM layer bearer capabilities are currently under study by ITU and the ATM Forum:

- Deterministic Bit Rate (DBR)
- Statistical Bit Rate (SBR) (or Variable Bit Rate (VBR)) with possible distinction between
  - SBR Real Time (SBR-RT)
  - SBR Non Real Time (SBR-NRT)
- ATM Block Transfer (ABT)
- Available Bit Rate (ABR)
- Unspecified Bit Rate (UBR).

Deterministic Bit Rate (DBR) is used by connections requiring a Peak Cell Rate (PCR) continuously available during the whole connection, typically to support Constant Bit Rate (CBR) applications. However, cells may be emitted either at or below the PCR. Statistical Bit Rate (SBR) connections are characterised by traffic parameters in addition to the PCR, namely Sustainable Cell Rate (SCR) with a Maximum Burst Size Tolerance limiting the maximum number of consecutive cells that may be sent at the PCR. The distinction between Real-Time SBR and Non-Real-Time is based on whether there are end-to-end cell delay variation commitments or not.

An ATM Block is a group of cells delineated by Resource Management (RM) cells. Using the ATM Block Transfer (ABT) bearer capability ATM Blocks are transmitted at a peak cell rate dynamically negotiated based on user requests by RM cells. There are two variants of this scheme, ABT with Delayed Transmission where the user is waiting on response from the network before cells are transmitted, and ABT with Immediate Transmission where the information is sent immediately after the request resource management cell. Available Bit Rate (ABR) can support applications that may adapt the information transfer rate to the currently available resources in the network based on feedback control messages from the network. The sources may increase the rate up to a specified maximum rate (PCR) and decrease down to a minimum usable cell rate (MCR). The Unspecified Bit Rate (UBR) is a best effort support for applications without any commitments from the network on QoS or transfer rate.

The most suitable choice of bearer capability depends on the specific service or application to support, however, there is not a one-to-one correspondence between services and bearer capabilities. Depending on the bearer capabilities to be supported different requirements will be posed on the ATM switches and the traffic control functionality.

The first ATM switches were developed with small buffers, typically in the order of 100 cell places per link, in order to obtain simple, common treatment of all service types in an integrated network. With small buffers in the switches the ATM cells will experience limited cell delay variation through the network, a favourable property for applications with strong real time requirements. Statistical multiplexing gain is possible with this kind of switches, i.e. additional traffic parameters (like SCR) are utilised to allow admission of more connections than when only using the peak cell rate, giving a statistical gain while still satisfying the QoS objectives. However, it is difficult to obtain high multiplexing gain for connections having high burstiness and high ratio between the peak cell rate and the link rate. For this type of connections it may be beneficial to have larger buffers in the switches, allowing bursts of cells to be stored, in order to achieve high network utilisation. This buffering causes higher variations in the cell transfer delays. Development of switches with large buffers typically started in the data communication industry making products for the LAN environment. ABT is another way to achieve statistical multiplexing gain for such sources by allocating the resources only when needed for transmitting the block of cells.

In order to satisfy both the wish for high network utilisation and the requirement on limited variation in cell transfer delays for some applications, there is a trend in the direction of developing switches with separate logical buffers for different types of bearer capabilities combined with service disciplines depending both on the connection type and the negotiated traffic parameters.

While switches with small buffers mainly have to rely on preventive traffic control schemes, the availability of larger buffers opens the possibility of making additional use of reactive congestion control mechanisms. This is of particular interest in LAN networks for applications that may adapt the information transfer rate to the resources in the network currently available. If these mechanisms are beneficial in a public ATM network with its geographical dimensions, widely different reaction times and complex meshed structure are being discussed.

Special Interest Group on Traffic Control and ATM Performance

ATM traffic control is of major importance to ensure a satisfying QoS to the customers, to secure that the network is robust towards overload and to achieve high network utilisation. A lot of activities are going on world wide to investigate these issues. The issues have been addressed in the RACE Programme through extensive theoretical investigations followed by experiments on ATM networks carried out in a lot of different projects. Within RACE a Special Interest Group (SIG) on Traffic Control and ATM Performance started in 1994 with the objectives of

- Identifying and analysing relevant topics in the field of traffic control and ATM performance, having reference to standardisation and specification activities
- Collecting information coming from the experimental and theoretical results produced by RACE projects
- Defining areas of common interest for new experiments
- Elaborating common position on hot topics in Traffic Control and ATM performance
- Contributing to Common Functional Specifications within RACE.

The main areas of interest for this SIG are:

- Definition and evaluation of multiplexing schemes
- Evaluation of mechanisms for the support of ABR
- Usage and Network Parameter Control (UPC/NPC) and traffic shaping
- Fast Resource Management (FRM, or ABT)
- ATM Performance.

The participating projects are BAF, BRAVE, COMBINE, EXPLOIT, LACE and TRIBUNE. The SIG produces a Baseline document on Traffic Control and ATM Performance [2] consisting of Part A: Theoretical results, and Part B: Experimental results, giving an overview of results obtained within the projects.

Here we will only give an indication of topics dealt with in these project related to the different ATM layer bearer capabilities, without the intention of giving an exhaustive description of all aspects covered.

EXPLOIT – Exploitation of an ATM Technology Testbed for Broadband Experiments and Applications

The EXPLOIT Testbed has been developed within two RACE projects with more than 30 partners (operators, industries and universities) involved. It is located in Basel and comprises:

- 4 ATM switches with various internal architecture
- Terminal equipment: Multimedia, TVs, PCs, N-ISDN
- Broadband signalling
- Interworking Units and Mappers
- Traffic generators and analysers.

The EXPLOIT testbed may be configured in a flexible way to create different complete ATM networks for traffic investigations. The ATM switches have small buffers and are used for studying the performance of preventive traffic control functions. A lot of experiments with different UPC algorithms have been carried out with main focus on combined peak cell rate and sustainable cell rate policing using the implemented dual leaky bucket mechanism. The performance of Connection Admission Control (CAC) algorithms have also been evaluated by multiplexing both homogeneous and heterogeneous traffic mixes with various on/off sources from traffic generators. The consistency and the robustness of the co-operation between the UPC and the CAC are being addressed. Thus, the statistical multiplexing gain obtainable based on the PCR and the SCR traffic parameters in a network with small buffers is evaluated. The small buffers can absorb only cell level congestion, hence the resource allocation scheme has to take care of the burst level statistics to prevent excessive cell losses. Furthermore, experiments are being carried out on resource management and performance assessment of end-to-end delays and cell losses through networks, including connections through the PEAN. Both artificial traffic from generators as well as real traffic applications are used in the experiments.

BRAVE – Broadband Access Verification Experiments

In the BRAVE project, multiplexing schemes based on FRM and ABR are investigated. The BRAVE platform for carrying out experiments to verify the chosen solutions consists of three broadband islands:

- one located in Milan and hosted by Italtel
- one hosted by Siemens (to be located in Berlin?)
- one located in Turin and hosted by CSELT.

These islands are interconnected through the Pan European ATM Network (PEAN).

The ABR protocol will be tested on this platform using an ATM PC connected to a Service Multiplexer (SM) and ATM Concentrator on the Siemens island, connected across the PEAN to

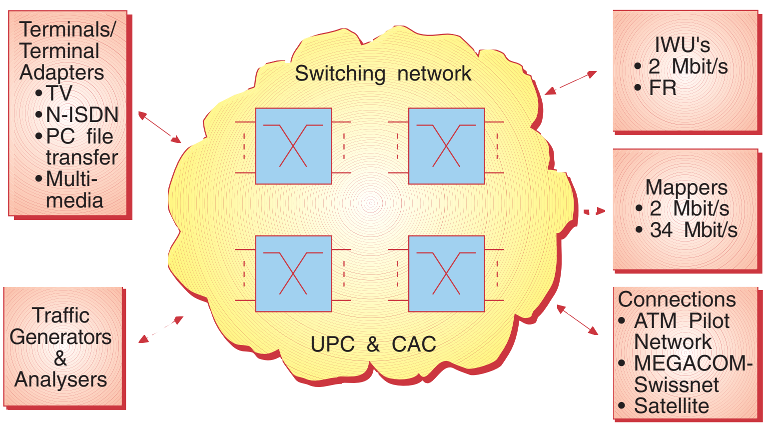


Figure 1 The EXPLOIT Testbed

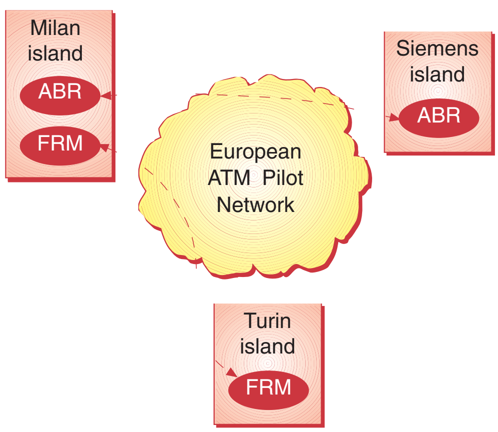


Figure 2 The BRAVE Platform

another SM with LANs at the Italtel island. A rate based ABR version is implemented in the terminal, the SM and the ATM Concentrator. Large buffers are necessary to cope with ABR traffic and will be implemented in the ATM Concentrator. The differences between how ATM traffic is handled when using ABR or VBR bearer capability will be highlighted in the experiments. Assessment of the multiplexing gain and the effects of ABR parameter choices on ABR performance will be given.

Furthermore, experiments will be performed with an FRM mechanism implemented in the Italtel and CSELT islands in the BRAVE platform interconnected via the PEAN. The efficiency of FRM schemes with delayed transmission depends heavily on the relation between the response time to reserve resources and the holding times of the resources necessary to transfer the information. The response time depends on the round-trip delay of the connection together with processing time in the FRM handling units. However, the holding times are determined by the traffic profile generated by the upper layers. In BRAVE the efficiency of FRM schemes will be studied by considering real traffic generated when running applications like file transfer and remote connections.

In FRM schemes there is a probability that the request from a user for allocation of resources will be rejected by the network due to lack of resources, and reattempts will be necessary after a certain delay. The burst level blocking probability becomes a QoS parameter in this scheme and will be measured when running real applications.

LACE – Local Applied Customer Environment for integrated communications

One way of supporting bursty data traffic is to base it on a best effort service. However, to be effective a loss-free operation should be ensured to avoid excessive retransmissions. Back-pressure policies to achieve this are under study in the LACE project. A hop-by-hop flow control will stop and restart the traffic per connection depending on the loading in the switch buffers. The mechanism will throttle the best effort traffic when congestion occurs to avoid cell losses in the network, and make it possible for this kind of traffic to grab as much as possible from the non-reserved and the unused resources. The efficiency in adapting to changing link load, the fairness independent of the location of sources, and the scalability with respect to number of nodes and distances are being studied.

References

1 ITU. *Traffic control and congestion control in B-ISDN.* Paris, 1995. (ITU-T Recommendation I.371, Frozen Issue.)

2 SIG on Traffic control and ATM performance, Baseline Document, Issue 2. 14 March 1995.

The PNO Cipher project

BY ØYVIND EILERTSEN

Background and motivation

The management systems of telecommunication networks and services are modelled using the concept of a Telecommunications Management Network (TMN). The security of these networks are obviously of fundamental importance for the operation of the networks. To study the security requirements of a pan-European TMN, EURESCOM set up the project "Security and Integrity Requirements" (EU-P110). This project started in October 1991 and was finished in December 1993. Among the key aspects that were recognised were

- confidentiality: the property that information is not made available or disclosed to unauthorised entities
- integrity: the property that data have not been altered or destroyed in an unauthorised manner.

The need for cryptographic mechanisms to achieve these goals became apparent. Therefore, a dedicated extension of the P110 project was set up to study the use of cryptography in a pan-European TMN. The study concluded with outlines for the key management and requirements for an encryption algorithm. The specification of these requirements are contained in Volume IV of the P110 final report.

Algorithm design was considered beyond the scope of EURESCOM, and the algorithm specification was delivered to ETSI with a request to use it for the design of a suitable algorithm.

A number of ETSI network operators responded positively to the idea of the development of such an algorithm. The ETSI Technical Assembly instructed the ETSI Security Techniques Advisory Group (STAG) to make recommendations for the development of an encryption algorithm; these can be found in the technical report prNA-TR 019 "Requirements specification for an encryption algorithm for operators of European public telecommunications networks." (September 1994) The specification was submitted to the ETSI Security Algorithms Expert Group (SAGE) for action.

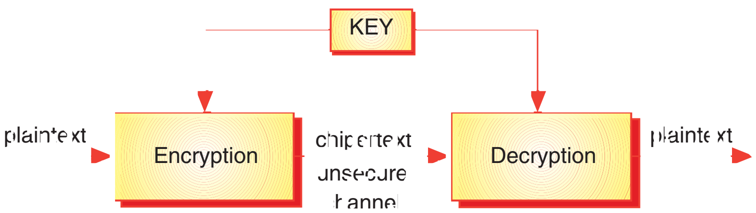


Figure 1

Algorithm usage

The users of the algorithm are network operators that are members of ETSI and licensed to operate a European public telecommunication network. Such an operator may also authorise other parties provided the restrictions listed below are not violated.

The algorithm may be used in both inter-domain (communication between separate domains) and intra-domain applications to ensure

- confidentiality and integrity protection of network management data
- authentication of entities used in the transfer of network management data.

In intra-domain applications, the algorithm may also be used for confidentiality and integrity protection of stored network management data.

Explicitly excluded uses are:

- Protection of communication between a user of services and a network operator, or between one user of such services and another
- Authentication of users, users' terminal equipment, or access to services provided by a network operator.

The restrictions on algorithm usage and the organisations entitled to use it are introduced to avoid some of the complicated legal matters that concern the use of cryptography in many European countries.

Functional requirements

The algorithm will be a symmetric cipher, i.e. the sender and receiver must have access to a common secret key that is used for both encryption and decryption. See illustration in Figure 1.

The algorithm will be a block cipher with block length 64 bits and key length options ranging from 64 to 128 bit in steps of 8 bit. The algorithm must be able to operate in the standard modes of operation for block ciphers defined by ISO in IS 10116. The algorithm should be easy to implement both in software and as a single-chip hardware device.

The design and complexity of the algorithm must facilitate both software and single-chip hardware implementations. A software implementation on a 32-bit microprocessor running at 25 MHz must be able to achieve a speed of at least 64 kbit/sec in standard Electronic Codebook Mode (cf. ISO 10116).

The operative lifetime of the algorithm should be at least 10 years, and the algorithm must in practice provide impenetrable protection throughout this period.

ETSI SAGE

The ETSI policy is to design tailor-made algorithms for specific applications. The algorithms are kept secret and distributed to member bodies on a "need-to-know" basis. SAGE (Security Algorithms Group of Experts) is a closed ETSI committee dedicated to the design of encryption algorithms. SAGE started its work in September 1991, and has designed several algorithms for use in ETSI standards.

The PNO cipher project

The encryption algorithm was dubbed the "PNO Cipher" and the official start of the design project (ETSI project PT70V) was February 1, 1995. The total manpower involved in the project is 34.5 man-months, of which Telenor Research contributes 12. According to the work plan, the project will be concluded August 1, 1996.

The project consists of five main tasks:

- Management
- Design
- Evaluation
- Algorithm usage
- Specification testing.

Telenor will be involved in all these tasks. The main part of our contribution will relate to the design, where Telenor contributes 8 out of a total of 15.5 man-months.

Glossary

ECB	Electronic Codebook Mode
ETSI	European Telecommunications Standardisation Organisation
EURESCOM	European Centre for Research and Strategic Studies in Telecommunications
IS	International Standard
ISO	International Standardisation Organisation
PNO	Public Network Operator
SAGE	Security Algorithms Expert Group
STAG	Security Techniques Advisory Group
TA	Technical Assembly
TMN	Telecommunications Management Network