DIMENSIONING OF IP ACCESS NETWORKS WITH ELASTIC TRAFFIC

Author Names: Anton Riedl*; Maren Perske; Thomas Bauschert; Andreas Probst
Company / Organization: Siemens AG; *Technische Universität München
Complete Mailing Address: Otto-Hahn-Ring 6, 81739 Munich, Germany
Telephone Number: +49 89 722 44056
Facsimile Number: +49 89 722 62230
Email Address: thomas.bauschert@icn.siemens.de

Theme: E. Revolution in Network Planning: Tools and Methods
Sub theme: 5. Designing for Varying Performance Levels

Abstract

Contrary to some announcements, bandwidth will remain a scarce resource in certain sectors such as Internet access via mobile networks. Thus, models for exact dimensioning and optimization of Internet access networks become vital for economy and quality of service. The Internet will soon integrate real-time (stream) and non-real-time (elastic) traffic classes. In this paper we address – as a first step of a larger, comprehensive procedure for both traffic classes - the dimensioning of IP access networks for elastic traffic. Our simulations confirm recently published theoretical achievements (processor sharing queueing models) and even prove the applicability of the dimensioning procedure onto tree-structured multi-link access systems.

1 Introduction

The Internet Protocol is becoming the basis of next generation networks. As a consequence, more and more systems are established to provide access to the Internet. While Internet access used to be bound to fixed networks (analog modems, ISDN, xDSL, or dedicated lines) new technologies like GPRS (General Packet Radio System) and UMTS (Universal Mobile Communication System) will provide Internet access for wireless and mobile systems. Especially for these systems, bandwidth will remain a precious good. But also in fixed networks with naturally higher and cheaper bandwidth offerings increasing competition makes exact dimensioning of capacities economically important, particularly for small and medium Internet access providers.

With regard to QoS requirements and traffic characteristics generally two Internet traffic classes are distinguished: stream and elastic. Stream traffic, which is generated by real-time applications such as audio or video communications, is usually transmitted via the RTP and UDP protocols and is sensitive to packet delay excess and packet delay variation. Elastic traffic refers to transmission of data where emphasis is laid on overall achieved throughput rather than individual packet delay. In the Internet, the TCP protocol is used for this type of traffic to adapt transfer rates to current network conditions. To achieve fair capacity sharing between active flows, TCP implements mechanisms like slow start and congestion avoidance. Combining elastic and stream traffic in a common infrastructure promises bandwidth savings. However, their different characteristics require certain control mechanisms like weighted fair queueing and admission control to guarantee respective QoS parameters.

The goal of this paper is to address – as a first step of a larger, comprehensive procedure for both traffic classes - the dimensioning of IP access networks for elastic traffic. In [1] the behavior of a single link fed by elastic traffic streams is described using processor sharing queueing models. Simulations shown in our paper confirm these theoretical achievements and even prove the applicability of the dimensioning procedure onto tree-structured multi-link access systems. The basic idea of the dimensioning procedure is to engineer the considered IP access network in a way that an average guaranteed transfer time is not exceeded. Outside the IP access network, throughput may be restricted due to limited customer access rates, e.g. at the air interface in mobile networks or due to insufficient server performance. This degradation is expressed by a delay factor specific to the access system. A prerequisite for our dimensioning method is to determine the maximum attainable throughput imposed from outside, e.g. by measurements. With these parameters the proposed procedure allows a quasi independent dimensioning of each link while still complying with the overall delay requirements. An interesting quality of the model is that it supports economy of scale effects, i.e. depending on the degree of aggregation higher utilization may be achieved on the aggregate link. This is also confirmed by our simulations.

2 Model

2.1 Network Model

We consider a hierarchical network architecture as shown in Figure 1. Internet customers are assigned to access areas respective to their location and are connected to the network through dedicated lines. Each access area is served by a Point of Presence (PoP) from where traffic is routed to an aggregation node. At this point, packets coming from several access areas are multiplexed onto one link and transferred to the backbone, either the provider's own or an upstream Internet Services Provider's (ISP) network. In the literature, the area between the first aggregation layer and the backbone layer is often called distribution layer. However, in our study there is no need to make this distinction. We focus on the star-structured network between the PoPs and the backbone and denote it, for short, as the access network.

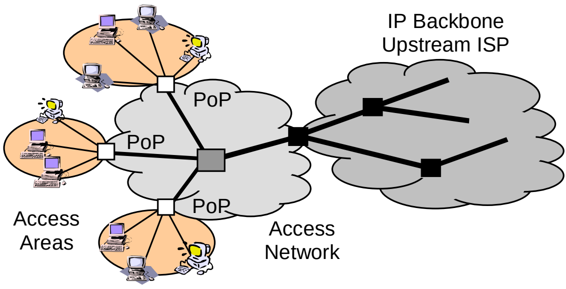

Figure 1: *Access network architecture*

2.2 Traffic Assumptions

In the following, only services, which generate elastic traffic are considered. Such services may imply the transmission of files containing data, text or picture, WWW pages and other documents.

Figure 2 presents the activity model, which these services fit into. A user who demands an Internet service starts an Internet session "dialing up" the Internet Service Provider. As long as the IP connectivity exists, various applications can be used to retrieve the desired information or to start a dialog. These applications usually set up several TCP connections – partially sequential and partially parallel – to one or more servers. The TCP protocol provides reliable, connection-oriented, and in-order service to them. During a TCP connection, data is transmitted in packets or bursts of packets across the network.

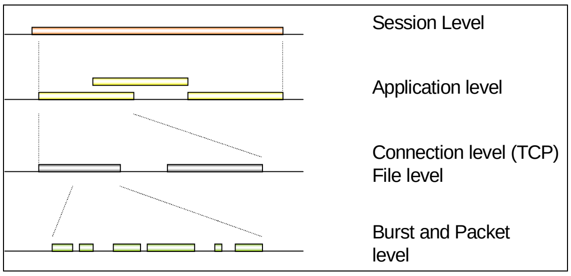

Figure 2: Activity model for services generating elastic traffic

The dimensioning procedure described in this paper is based on IP traffic characterization at TCP connection level. We assume that every elastic traffic flow is related to a single file transfer. Furthermore, new file transfers start according to a Poisson process and file sizes are heavy-tail distributed, e.g. modeled by a Pareto distribution. However, the presented formulas can be applied to any file size distribution as they are not sensitive to the shape of the general service time distribution. Furthermore, for simplification, we assume that all Internet users experience approximately the same maximum access bit rate (due to similar modem connections to the access node).

The rate at which sources are allowed to send traffic into the network is controlled by TCP as a function of network congestion. The TCP feedback mechanism is assumed to be ‘ideal’ (i.e. instantaneous feedback), therefore all elastic traffic flows going over one link share the link capacity equally.

2.3 Single Link Model

In this section we address the queueing behavior of elastic connections entering a single network link without admission control.

Assuming that the TCP feedback mechanism is ideal, a network link carrying only elastic traffic flows can be modeled by a processor sharing queue (see [1]-[6]). One can imagine that several files that need to be transmitted over one link are broken up into little pieces, i.e. the individual IP packets of the different traffic flows, and are processed by the link quasi-simultaneously. In case the attainable transfer rate for one connection is limited to a certain peak rate r_{peak}, the link appears like a system with R servers where $R=C/r_{peak}$, i.e. the ratio between the link rate C and the peak source rate r_{peak} . Up to R flows can be served at the same time without rate reduction imposed by the system. If r_{peak} is larger than the link capacity the M/G/R-PS model reduces to the simple M/G/1-PS model. A nice property of M/G/R-PS queues is the independence of the average sojourn time (average time in system) from the shape of the general service time or file size distribution. Large files do not delay small ones too much.

For the M/G/R-PS model, the expected sojourn time (or transfer time) $E\{T(x)\}$ for a file of size x is given by:

$$E\{T(x)\} = \frac{x}{r_{peak}}\left(1+\frac{E_2(R,R\rho)}{R(1-\rho)}\right) = \frac{x}{r_{peak}}\cdot f_R \qquad \text{Equation 1}$$

Consequently, the actual average bit rate (throughput) D during the file transfer phase is:

$$D = \frac{r_{peak}}{\left(1+\frac{E_2(R,R\rho)}{R(1-\rho)}\right)} \qquad \text{Equation 2}$$

Here, ρ denotes the utilization on the link ($\rho = \lambda_e \cdot x_{mean}/C$ with flow arrival rate λ_e and average file size x_{mean}). E_2 represents Erlang’s second formula (Erlang C formula) with $A = R\rho$:

$$E_2(R,A)=\frac{\frac{A^R}{R!}\cdot\frac{R}{R-A}}{\sum_{i=0}^{R-1}\frac{A^i}{i!}+\frac{A^R}{R!}\cdot\frac{R}{R-A}}$$

Equation 3

In Equation 1 the expression in parentheses is also called ”delay factor“ f_R as it denotes the increase of the average file transaction time imposed by the link. The delay factor is a quantitative measure of how link congestion affects transaction times, taking into account the economy of scale effect.

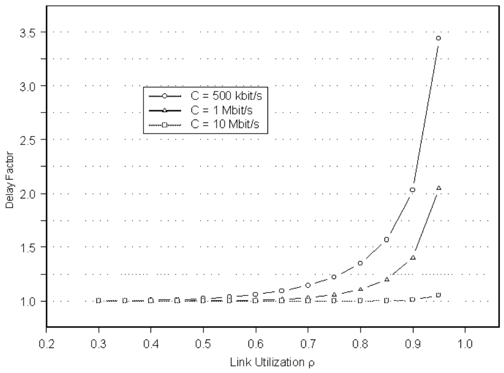

Figure 3: Delay Factor dependent on link utilization and link capacity

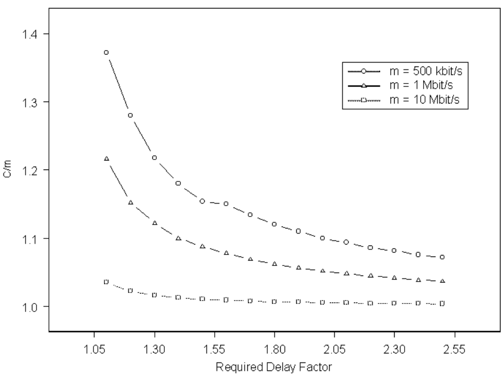

Figure 4: Overprovision factor dependent on required delay factor

Figure 3 illustrates the dependency between the delay factor and the link utilization for different degrees of aggregation. As one can see, links with larger capacities can achieve higher utilization levels than small links while still maintaining low delay factors. Figure 4 shows the same relationship just from a link dimensioning point of view. Assuming a certain required delay factor, the graph gives the necessary capacity C as a multiple of the mean bit rate $m = \lambda_e x_{mean}$. In both graphs, r_{peak} is set to 64 kbit/s.

3 Dimensioning Procedure

Our dimensioning procedure for elastic traffic considers the average file transfer time (for files of a certain size) as the relevant QoS criterion. This time should be restricted by some upper bound respective to a typical Internet user’s assumed tolerance. From another perspective, the dimensioning could be driven by an average throughput requirement for each flow.

First we consider the more simplistic case of single link dimensioning. As mentioned above, there are two slightly different approaches:

(1) First, the objective is to guarantee a certain average transfer time $E\{T(x)\}$ for a file of size x. Of course, this value has to be chosen greater than the minimum possible transfer time x/r_{peak}. With given x and $E\{T(x)\}$ we can resolve Equation 1 numerically to get the desired capacity value.

(2) The second approach considers a certain average throughput guarantee for all file transactions. Note that this throughput value must be smaller than the maximum access bit rate r_{peak}. According to Equation 2 the throughput (bit rate) during a file transaction turns out to be r_{peak} / f_R and is independent of the file size x. Consequently, all file transactions suffer the same average throughput degradation. However, this only holds if all transactions are restricted by the same limited access rate r_{peak}. Again, solving Equation 2 gives the desired capacity value.

Now, we extend the procedure for dimensioning of multiple concatenated links or whole networks:

In a network the average file transfer time or the average throughput has to be guaranteed on an end-to-end basis. As shown in our simulations (see next chapter) the end-to-end QoS criteria are sufficiently met if each link is dimensioned according to this requirement. A possible explanation is as follows: due to the TCP control mechanism the whole network can be regarded as a single processor sharing system. For an end-to-end connection, which runs over multiple network links only the bottleneck link (i.e. the link with the highest delay factor f_R) determines the transaction bit rate (r_{peak} / f_R) and, thus, the file transfer time (indeed the other links have only marginal influence). Consequently, optimum network dimensioning means assigning link capacities in a way that each link suffers the same delay factor (which in turn is driven by the end-to-end QoS requirement). This dimensioning procedure neglects the dependencies in the behavior of concatenated links. However, simulations show that for practical delay factors (which are close to 1) our dimensioning procedure works well despite the independence assumption.

It should be noted that the sojourn time formula (Equation 1) does not take into account the explicit queueing delay of individual IP packets of the TCP connections as the queueing entities are the TCP connections themselves. However, assuming an ideal (instantaneous) operation of the TCP control mechanism as well as a proper link dimensioning (i.e. no capacity underprovisioning which may lead to high delay factors) each incoming IP packet is forwarded almost instantaneously and suffers only a negligible delay (e.g. 0.04 s for an IP packet of 552 byte length and an output rate of 100 kbps) compared to typical file transfer times.

For the link model in section 2.3 it was assumed that all Internet access users have approximately the same maximum access bit rate r_{peak}. This holds true today where the majority of the users are still connected via modems to the Internet. However, due to the evolving high-speed Internet access technologies the situation where different file transactions have different highest possible bit rates becomes more likely. In this case we either have to find a multirate extension of Equation 1 (like in the case of the multirate Erlang-B formula) or reuse the single rate formula in a proper way. The first possibility is for further study whereas in the second case we can proceed in two directions: 1) from the different r_{peak} calculate an average value (e.g., $r_{peak\_average} = \rho^{-1} \sum \rho_i r_{peak\_i}$) and use this average value for dimensioning; 2) define traffic classes with equal r_{peak} value (e.g. for each access technology like 64k ISDN, 2M SDSL etc.) and calculate the required capacities for each traffic class separately. Finally sum up the capacities. This certainly results in some overdimensioning as traffic integration savings are not considered.

4 Dimensioning Example and Simulation Results

In order to examine the validity of the model, we apply our proposed algorithm to the access network scenario depicted in Figure 5 and compare the theoretical results with simulation data. To allow conclusions about the basic applicability of the M/G/R-PS model to tandem links in conjunction with the TCP protocol, we stick closely to the assumptions in section 2.2 and keep topology as well as traffic parameters simple. Thus, the traffic characteristics in all access areas are assumed to be similar. In our simulations, we generate an average bit rate of about 94 kbps per access area (i.e. on each access links). The traffic enters the access network at the backbone node and traverses the aggregation link and the respective access link. This gives an average bit rate of 940 kbps on the aggregation link. The file size is Pareto distributed with shape parameter 1.5 and mean value 12 kbytes. Access speeds (r_{peak}) are assumed to be 64 kbps. For a file size of 100 kbytes (including TCP/IP overhead) an average transaction time of at most 15 seconds is demanded. Since at 64 kbps access speed the minimum transfer time of such a file is already 12.5 seconds, the delay factor should be below or equal to 1.2. Applying the suggested dimensioning procedure, we obtain capacity values of 192 kbps for access links and 1.088 Mbps for the aggregation link.

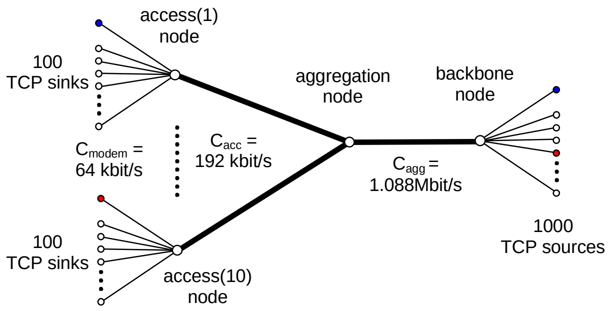

Figure 5: Access network scenario used in our simulations

For our simulations we use the network simulator ns available from Berkeley Laboratories [7]. As TCP protocol we choose the FullTCP module which implements mechanisms for slow-start and congestion avoidance. A total number of 1000 TCP sink nodes represent the customers and are assigned to the 10 access nodes evenly. The ten access nodes are connected to the aggregation node from where a link is set up to the backbone node. The TCP sources are directly connected to the backbone nodes, thus representing servers, actually proxy servers, in the Internet. Every TCP source corresponds to a TCP sink. Files are generated at the sources and are sent to the respective sinks. All links are full duplex with equal capacity in both directions. However, we simulated only the downstream data path since it gives us a first impression of how well our model works for the TCP capacity sharing mechanism.

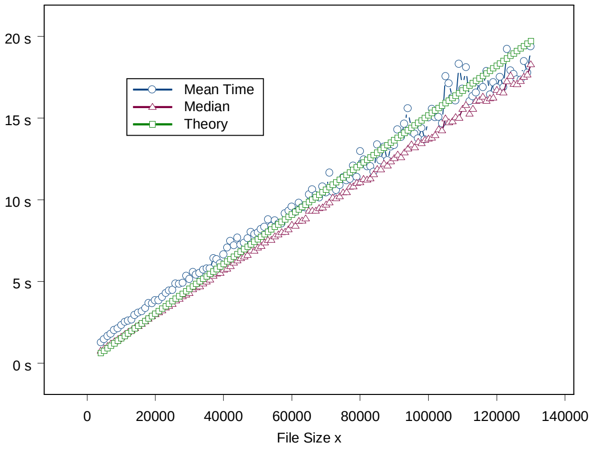

Figure 6 illustrates a typical simulation result. File transaction times (mean and median values) are plotted over the file size *x*. In addition, the theoretical values according to Equation 1 are given. As we can see, small files experience a somewhat larger average transaction time than computed by Equation 1. For increasing file sizes the theoretical model becomes more accurate and, finally, even overestimates the transaction time.

Figure 6: Transaction time vs. file size (Bytes)

This behavior has been shown for all simulations with different access speeds, traffic intensities, and mean file sizes. A possible interpretation is that large files have a long enough transaction time for them to see the average behavior of the whole system, i.e. the access network. Furthermore, long flows seem to take advantage of the TCP control mechanism’s inability to react to a new flow instantaneously. Therefore, the bandwidth of active flows is not reduced as soon as a new flow enters the system, which gives long files a little higher share of the available capacity. Small files, on the other side, are slightly disadvantaged and exceed their delay requirement. However, since their absolute mean transaction time is small, the increase does not affect the customers’ perception of quality. It seems more important that absolute transaction times, which come close to the users’ tolerance level are kept under a certain limit.

5 Conclusion

The M/G/R-PS model, which we propose as a basis for dimensioning of IP access networks works well for TCP controlled elastic traffic in a homogenous environment. Our simulations confirm its basic applicability, especially in the case of transactions of large files. However, further studies are necessary and under way to extend the approach for networks with different access speeds, unbalanced traffic requirements, or varying propagation delays. Furthermore, it needs to be examined how various notions of TCP implementations affect the behavior.

References

[1] K. Lindberger, Balancing Quality of Service, Pricing and Utilisation in Multiservice Networks with Stream and Elastic Traffic, ITC 16, Edinburgh, 1999

[2] L. Kleinrock, Queueing Systems Volume II: Computer Applications, Wiley-Interscience Publication, 1976

[3] A.P. Zwart, O.J. Boxma, Sojourn time asymptotics in the M/G/1 processor sharing queue, Centrum voor Wiskunde en Informatica (CWI), PNA-R9802 April 30, 1998

[4] R. Núnez Queija, J.L. van den Berg, M.R.H. Mandjes, Performance evaluation of strategies for integration of elastic and stream traffic, Centrum voor Wiskunde en Informatica (CWI), PNA-R9903 February 28, 1999

[5] S. Borst, O. Boxma, P. Jelenkovic, Generalized Processor Sharing with Long-Tailed Traffic Sources, ITC 16, Edinburgh, 1999

[6] J.W. Cohen, The Multiple Phase Service Network with Generalized Processor Sharing, Acta Informatica 12, 1979

[7] Network simulator ns, http://www-mash.cs.berkeley.edu/ns