REPORTRAPPORT

PNA

Probability, Networks and Algorithms

Probability, Networks and Algorithms

Smart dimensioning of IP network links

R. van de Meent, M.R.H. Mandjes, A. Pras

Centrum voor Wiskunde en Informatica (CWI) is the national research institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organisation for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

Smart dimensioning of IP network links

ABSTRACT

Link dimensioning is generally considered as an effective and (operationally) simple mechanism to meet (given) performance requirements. In practice, the required link capacity C is often estimated by rules of thumb, such as C=dM, where M is the (envisaged) average traffic rate, and d some (empirically determined) constant larger than 1. This paper studies the viability of this class of `simplistic' dimensioning rules. Throughout, the performance criterion imposed is that the fraction of intervals of length T in which the input exceeds the available output capacity (i.e., CT) should not exceed epsilon, for given T and epsilon. We first present a dimensioning formula that expresses the required link capacity as a function of M and a variance term V(T), which captures the burstiness on timescale T. We explain how M and V(T) can be estimated with low measurement effort. The dimensioning formula is then used to validate dimensioning rules of the type C=dM. Our main findings are: (i) the factor d is strongly affected by the nature of the traffic, the level of aggregation, and the network infrastructure; if these conditions are more or less constant, one could empirically determine d; (ii)we can explicitly characterize how d is affected by the "performance parameters", i.e., T and epsilon.

Smart Dimensioning of IP Network Links

Remco van de Meent[1], Michel Mandjes[2], and Aiko Pras[1]

[1] University of Twente, Netherlands
{r.vandemeent,a.pras}@utwente.nl
[2] University of Amsterdam and Centre for Mathematics and Computer Science,
Netherlands
mmandjes@science.uva.nl

Abstract. Link dimensioning is generally considered as an effective and
(operationally) simple mechanism to meet (given) performance require-
ments. In practice, the required link capacity C is often estimated by
rules of thumb, such as $C = d \cdot M$, where M is the (envisaged) average
traffic rate, and d some (empirically determined) constant larger than 1.
This paper studies the viability of this class of 'simplistic' dimensioning
rules. Throughout, the performance criterion imposed is that the fraction
of intervals of length T in which the input exceeds the available output
capacity (i.e., CT) should not exceed ε, for given T and ε.

We first present a dimensioning formula that expresses the required link
capacity as a function of M and a variance term $V(T)$, which captures the
burstiness on timescale T. We explain how M and $V(T)$ can be estimated
with low measurement effort. The dimensioning formula is then used to
validate dimensioning rules of the type $C = d \cdot M$. Our main findings
are: (i) the factor d is strongly affected by the nature of the traffic, the
level of aggregation, and the network infrastructure; if these conditions
are more or less constant, one could empirically determine d; (ii) we can
explicitly characterize how d is affected by the 'performance parameters',
i.e., T and ε.

1 Introduction

In order to meet the users' performance requirements on an Internet connection,
two approaches seem viable, see, e.g., [1,2]. The first approach relies on the use
of protocols that enforce certain service levels, for instance by prioritizing some
streams over other streams, by performing admission control, or by explicitly
dedicating resources to connections; examples of such techniques are DiffServ
[3] and IntServ [4]. The second approach does not use any traffic management
mechanisms, but rather relies on allocating sufficient network capacity to the
aggregate traffic stream. In this approach the link capacity should be chosen
such that it is always large enough to satisfy the performance requirements of
all flows. This approach, which is often called *overdimensioning*, is commonly
used by network operators for their backbone links; some studies found that such
links generally have a capacity which is '30 times the average traffic rate' [5].

As described in, e.g., [6,7], it has several advantages to guarantee the users'
performance requirements (agreed upon in a service level agreement, or SLA)

by relying on link dimensioning. Perhaps the most significant advantage is that dimensioning is (operationally) *simple*; it eliminates the need for network systems and network management to support relatively complex (and therefore error-prone) techniques for enforcing the SLA parameters.

Although the idea of link dimensioning is simple, still the question remains of *how much* link capacity is needed to guarantee the parameters agreed upon in the SLA. Without sufficient capacity, the performance, as experienced by the users, will drop below the required levels. If the link is dimensioned too generously, however, then the performance does not improve anymore, and hence resources are essentially wasted. This trade-off leads to the concept of *smart dimensioning*, which we define as the lowest link capacity at which the SLA is met.

When determining this link capacity, a specific question is for instance: is there, for a given performance target, a fixed ratio between the required capacity and the *average* traffic rate? If there would be, then we would evidently have a simple and powerful dimensioning rule. A more detailed question concerns the dependence of d on the performance requirement imposed: when making the performance target more stringent, evidently d should increase, but can this dependence be quantified?

Approach and organization. The idea in this paper is to study smart dimensioning, as introduced above; the main question is 'what is the link capacity that is minimally required?' Throughout, the performance criterion imposed is that the fraction of intervals of length T in which the input exceeds the available output capacity (i.e., CT) should not exceed ε, for given T and ε.

There are various possible approaches to answer this question. For instance, one could follow a fully empirical approach. Then one experimentally increases (or decreases) a network link's link capacity, and evaluates the performance as experienced by the users, so as to determine the minimally required link capacity.

We opt, however, for a different approach: we first derive an analytical link dimensioning formula; this gives the required link capacity to achieve a certain performance target, for given input traffic (in term of a mean rate and a variance term that expresses the traffic aggregate's burstiness). Then we explain how these traffic parameters can be estimated with minimal measurement effort. We prefer this approach, mainly because of its *systematic* nature: it explicitly shows which parameters of the underlying traffic process essentially determine the required link capacity, and how it is affected by the performance requirement.

The present paper builds upon previous work on traffic modeling and network link dimensioning [8,9,11,12]. Section 2 recapitulates our findings on the modeling of real network traffic (based on our measurements at 5 representative networking environments); importantly, these measurements indicate that under fairly general circumstances the Gaussian traffic model applies. We also derive a link dimensioning formula, which greatly simplifies under Gaussianity; this formula shows how the 'performance parameters' T and ε affect the required link capacity. Section 3 reviews approaches to estimate the Gaussian traffic model's parameters, i.e., mean and variance. In Section 4 it is discussed how to apply the link dimensioning formula from Section 2 in practice, through an evaluation of its performance in different scenarios. Section 5 systematically assesses

the amount of link capacity required; interestingly, it is also shown how one could explicitly predict the impact of changing T and/or ε on the required link capacity. Concluding remarks are provided in Section 6.

2 Link dimensioning formula

As argued in the introduction, an important prerequisite for dimensioning are formulas that determine, for given characteristics of the offered traffic and performance target, the minimum required link rate. Preferably, these dimensioning formulas have minimal requirements on the 'nature' of the traffic offered; for instance, we do not want to impose any conditions on its correlation structure. In this section, we present a formula that relies on only weak conditions on the traffic process, i.e., stationarity and Gaussianity:

- *Stationarity* means that, with $A(s,t)$ denoting the amount of traffic arrived in the time interval $[s,t)$, the distribution of $A(s+\delta, t+\delta)$ does not depend on δ (but just on the interval length $t-s$). In the sequel we use the abbreviation $A(t) := A(0,t)$.
- *Gaussianity* refers to the probability distribution of $A(t)$. It is supposed that $A(\cdot)$ is a Gaussian process with stationary increments, i.e., $A(s,t)$ is normally distributed, with mean $M \cdot (t-s)$ and variance $V(t-s)$, for some mean rate $M \in \mathbb{R}$ and variance curve $V(\cdot) : \mathbb{R}^+ \to \mathbb{R}^+$.

Stationarity is a common assumption in traffic modeling; it usually applies on timescales up to, say, hours. In earlier work, we have thoroughly investigated the Gaussianity of real Internet traffic, in various representative settings (in terms of types of users, network infrastructure, timescales, etc.) — see, e.g., [8,9]. We found that a Gaussian traffic model accurately describes real traffic, particularly when the level of aggregation was sufficiently high. We note that this Gaussianity issue was the subject of a number of other studies, see for instance Fraleigh *et al.* [6] and Kilpi and Norros [10]; there similar conclusions were drawn.

Derivation of link dimensioning formula for Gaussian traffic. Given the observation that a real Internet traffic stream can be accurately approximated by a Gaussian process, we now develop a formula that estimates the minimally required link capacity to cater for that traffic stream.

First, however, we specify what 'to cater for a traffic stream' means. In this paper we rely on the notion of *link transparency* that was introduced in [11]. Its main objective is to ensure that the links are more or less 'transparent' to the users, in that the users should not (or almost never) perceive any performance degradation due to a lack of bandwidth. Clearly, this objective will be achieved when the link rate is chosen such that only during a small fraction of time ε the aggregate rate of the offered traffic (measured on a sufficiently small time scale T) exceeds the link rate: $\mathbb{P}(A(T) \geq CT) \leq \varepsilon$. The values to be chosen for the parameters T and ε typically depend on the specific needs of the application(s) involved. Clearly, the more interactive the application, the smaller T and ε should be chosen; network operators should choose them in line with the SLAs they agreed upon with their clients.

Now, given the criterion $\mathbb{P}(A(T) \geq CT) \leq \varepsilon$, we can derive a formula for the minimal link rate needed (without assuming Gaussian input at this point). Relying on the Markov inequality $\mathbb{P}(X \geq a) \leq \mathbb{E}(X)/a$ for a non-negative random variable X, we have for $\theta \geq 0$ that $\mathbb{P}(A(T) \geq CT) \leq \mathbb{E} \exp(\theta A(T)) \exp(-\theta CT)$, and hence we obtain the celebrated Chernoff bound

$$\mathbb{P}(A(T) \geq CT) \leq \min_{\theta \geq 0} \left(e^{-\theta CT} \mathbb{E} e^{\theta A(T)} \right).$$

Rewriting this expression, it is not hard to see that, in order to be sure that $\mathbb{P}(A(T) \geq CT) \leq \varepsilon$ it suffices to take the link's bandwidth capacity C at least

$$C \equiv C(T, \varepsilon) = \min_{\theta \geq 0} \frac{\log \mathbb{E} \exp(\theta A(T)) - \log \varepsilon}{\theta T}. \tag{1}$$

Finally, imposing some additional structure on $A(\cdot)$ simplifies the general dimensioning formula of (1). When assuming traffic is Gaussian, with $\delta := \sqrt{-2 \log \varepsilon}$, the dimensioning formula (1) reduces to

$$C = M + \frac{\delta}{T} \cdot \sqrt{V(T)}; \tag{2}$$

here it is used that $\mathbb{E} \exp(\theta A(t)) = M\theta t + \theta^2 V(t)/2$. The important consequence of this, is that for the application of the dimensioning formula (2) in this Gaussian context it is required to have estimates for the mean rate M and the variance $V(T)$.

3   Estimating traffic parameters

In the previous section we concluded that, in order to dimension a network link by applying dimensioning formula (2), an accurate estimate of the traffic offered (both in terms of the mean traffic rate M, as well as its fluctuations, expressed through $V(T)$) is required. Estimating M is relatively straightforward, and can be done through standard coarse traffic measurements, e.g., by polling Interfaces Group MIB counters via SNMP (Simple Network Management Protocol) every 5 minutes.

Estimating the variance $V(T)$ (which could be interpreted as 'burstiness'), however, could be substantially harder: particularly on smaller timescales T, it is hard to do accurate measurements through SNMP. The standard way to estimate $V(T)$ (for some given small interval length T) is what we refer to as the 'direct approach': perform traffic measurements for disjoint intervals of length T, say $a_i(T)$ for $i = 1, \ldots, N$, and compute their sample variance

$$(N-1)^{-1} \sum_{i=1}^{N} (a_i(T) - MT)^2.$$

An important drawback to this direct approach, however, is that it requires substantial measurement effort to accurately measure $a_i(T)$ for small T. This

Location	Short description	# traces	Mean rate (Mbit/s)
U	university residential network (1800 hosts)	15	170
R	research institute (250 hosts)	185	6
C	college network (1500 hosts)	302	35
A	ADSL access network (2000 hosts)	50	120
S	server hosting provider (100 hosts)	201	12

Table 1. Measurement locations

drawback is countered by our so-called 'indirect approach', which is briefly discussed next — we refer to [12] for an in-depth description.

Indirect estimation of $V(T)$. The 'indirect approach' to estimate $V(T)$ relies on (coarse-grained) measurements of the buffer occupancy, as follows. By regular polling the occupancy B of the buffer in front of the to-be-dimensioned network link, the probability distribution $\mathbb{P}(Q > B)$ of the buffer occupancy is estimated. Interestingly, as shown in [12], for Gaussian inputs, the distribution of the buffer occupancy uniquely determines the variance function $V(\cdot)$ of the input process, for given mean rate M; in particular, it was shown that it does so through the following relation:

$$V(t) \approx \inf_{B \geq 0} \frac{(B + (C - M)t)^2}{-2 \log \mathbb{P}(Q > B)}.$$

In other words: when knowing $\mathbb{P}(Q > B)$ (or an accurate estimate), we can infer $V(t)$ for any timescale t. As our numerical and experimental evaluation in [12] shows, the above 'indirect approach' to obtain $V(\cdot)$ from coarse-grained measurements, yields estimates of the variance that are remarkably close to the actual values.

Hence, we can estimate both M and $V(T)$ with relatively low measurement effort. In the next section we demonstrate how these can be used to support finding an accurate estimate of the required link capacity.

4 Dimensioning

In Section 2 we developed a link dimensioning formula (2) for Gaussian network traffic, which has the input parameters the mean M and variance $V(T)$, and is supposed to meet the performance target $\mathbb{P}(A(T) \geq CT) \leq \varepsilon$. In Section 3 we then explained how M and $V(T)$ could be estimated through coarse measurements. In the present section, the estimates of M and $V(T)$ are inserted into the dimensioning formula (2) to estimate the minimally required link capacity. We can then verify whether the performance criterion imposed is actually met. We will do so through a number of case studies — a sizable collection of traffic traces of 15 minutes each, from various representative locations, see Table 1; for more detailed information, see [9, Section 2.3].

We evaluate the accuracy of the dimensioning formula (2). It requires knowledge of M and $V(T)$, which we estimate as described in Section 3; in particular,

$V(T)$ is estimated through the 'indirect approach'. This indirect approach requires an estimate of $\mathbb{P}(Q > B)$ (as a function of $B \geq 0$); this was enabled by a simple simulation environment that 'replays' the real traffic trace through a simulated buffer and link. The resulting estimates are inserted into (2), yielding the estimated minimally required link capacity for a chosen ε and T. In the present experiments, we set ε to 1%, and set T to 1 sec, 500 msec and 100 msec. These are timescales that are, for various applications, important to the perception of quality by (human) users, and thus are relevant when striving for link transparency. Now it is interesting to validate whether, under the estimated minimally required link rate, the performance requirement would be met.
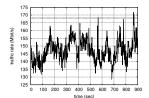
A first validation result is presented in Fig. 1. It shows the estimated required bandwidth for three different values of T, with $\varepsilon = 0.01$, for location A. It is noted that the fluctuations of the traffic rate in this specific example are relatively low compared to the mean traffic rate. This is because at this location a large number of relatively small (ADSL) access links are multiplexed on a large (1 Gbit/sec) backbone, and therefore a single user cannot have a strong impact on the aggregate traffic stream.
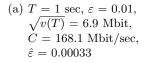
Because of the rather small fluctuations, the amount of extra bandwidth required to cater for the peak traffic rates (which is desirable under the link transparency criterion imposed), compared to the mean traffic rate, is also relatively small: some 20% at the 100 msec timescale. Later on in this paper we will see that in other scenarios, the extra required bandwidth can be as high as hundreds of percents.
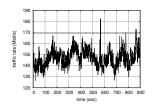
Figs. 2 and 3 present similar results for locations U and S, respectively. Fig. 2 shows an interesting example of a heavily loaded network: it can be shown that the peak traffic rates in this example trace, even at small timescales, are lower than may be expected from a Gaussian traffic stream with the estimated mean and variance. As a result of this, the 'realized performance' (in terms of the $\hat{\varepsilon}$ that will be defined below) is well below the anticipated $\varepsilon = 0.01$. This might be caused by the relatively high average traffic rate (compared to the other parts in this same trace), from the approximately 280 th to 420 th second.

Fig. 3 illustrates the importance of looking at small timescales when dimensioning network links: the peak rates at small timescales, in this particular example, are sometimes as much as 6 times the average traffic rate. Evidently, also the setting of ε is of importance when determining the required bandwidth capacity. It can clearly be seen from Fig. 3 than when ε is set smaller than the 0.01 chosen here, the estimated required bandwidth capacity increases significantly, as then a larger number of the traffic peaks should be catered for.
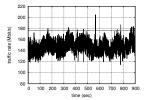
The above experiments already gave a rough impression about the performance of our dimensioning procedure. In order to further validate how well the estimated bandwidth capacity C corresponds to the required bandwidth, we introduce the notion of 'realized exceedance', denoted with $\hat{\varepsilon}$. We define the 'realized exceedance' as the fraction of (disjoint) intervals of length T, in which the amount of offered traffic $a_i(T)$ exceeds the estimated required capacity CT — we stress the fact that 'exceedance' in this context does not correspond to 'packet
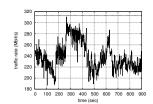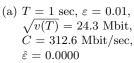
(a) $T = 1$ sec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 6.9$ Mbit, $C = 168.1$ Mbit/sec, $\hat{\varepsilon} = 0.00033$

(b) $T = 500$ msec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 3.6$ Mbit, $C = 169.4$ Mbit/sec, $\hat{\varepsilon} = 0.0028$
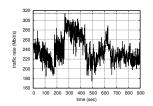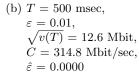
(c) $T = 100$ msec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 0.97$ Mbit, $C = 176.6$ Mbit/sec, $\hat{\varepsilon} = 0.00027$
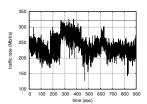
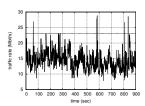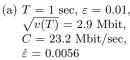**Fig. 1.** Case-study for location A, example trace with ($M = 147$ Mbit/sec)

(a) $T = 1$ sec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 24.3$ Mbit, $C = 312.6$ Mbit/sec, $\hat{\varepsilon} = 0.0000$

(b) $T = 500$ msec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 12.6$ Mbit, $C = 314.8$ Mbit/sec, $\hat{\varepsilon} = 0.0000$

(c) $T = 100$ msec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 2.7$ Mbit, $C = 320.8$ Mbit/sec, $\hat{\varepsilon} = 0.0003$

Fig. 2. Case-study for location U, example trace with ($M = 239$ Mbit/sec)

(a) $T = 1$ sec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 2.9$ Mbit, $C = 23.2$ Mbit/sec, $\hat{\varepsilon} = 0.0056$

(b) $T = 500$ msec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 1.6$ Mbit, $C = 24.3$ Mbit/sec, $\hat{\varepsilon} = 0.0083$

(c) $T = 100$ msec, $\varepsilon = 0.01$, $\sqrt{v(T)} = 0.45$ Mbit, $C = 27.8$ Mbit/sec, $\hat{\varepsilon} = 0.0100$

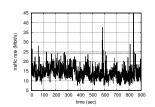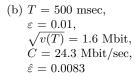**Fig. 3.** Case-study for location S, example trace with ($M = 14.3$ Mbit/sec)

loss'. In other words:

$$\hat{\varepsilon} \equiv \hat{\varepsilon}(C) := \frac{\#\big\{i \in \{1,\ldots,N\} \mid a_i(T) > CT\big\}}{N}.$$

If C is properly estimated, then 'exceedance' (as in $a_i(T) > CT$) may be expected in a fraction ε of all intervals. There are, however, (at least) two reasons why $\hat{\varepsilon}$ and ε may not be equal in practice. (i) Firstly, (2) assumes 'perfectly Gaussian' traffic, which is not always the case [8]. Evidently, deviations of 'perfectly Gaussian' traffic may have an impact on the estimated C. (ii) Secondly, to obtain (1), an upper bound (viz. the Chernoff bound) on the target probability has been used, and it is not clear upfront how far off this bound is.

Location	T	avg. $\lvert\varepsilon - \hat{\varepsilon}\rvert$	stderr $\lvert\varepsilon - \hat{\varepsilon}\rvert$
U	1 sec	0.0095	0.0067
	500 msec	0.0089	0.0067
	100 msec	0.0077	0.0047
R	1 sec	0.0062	0.0060
	500 msec	0.0063	0.0064
	100 msec	0.0050	0.0053
C	1 sec	0.0069	0.0047
	500 msec	0.0066	0.0043
	100 msec	0.0055	0.0041
A	1 sec	0.0083	0.0027
	500 msec	0.0083	0.0024
	100 msec	0.0079	0.0020
S	1 sec	0.0052	0.0050
	500 msec	0.0049	0.0055
	100 msec	0.0040	0.0059

Table 2. Required bandwidth: estimation errors ($\varepsilon = 0.01$).

To assess to what extent the dimensioning formula for Gaussian traffic is accurate for real traffic, we compare ε and $\hat{\varepsilon}$. We do this comparison for the hundreds of traces that we collected at measurement locations $\{U, R, C, A, S\}$. Table 2 presents the average differences between the targeted ε and the 'realized exceedance' $\hat{\varepsilon}$ at each location (where the averaging is done over all traces collected at that location), as well as the corresponding standard deviations, for three different timescales T (1 sec, 500 msec and 100 msec). The table shows that differences between ε and $\hat{\varepsilon}$ are small. Hence, we conclude that our approach accurately estimates the required bandwidth to meet the pre-specified performance target.

5 Dimensioning factors

In this section we address the question whether there is, for a given performance target, a fixed ratio between the required capacity C and the average traffic rate M. We start this section, however, with a quantification of this ratio as a function of the parameters T and ε (i.e., the parameters that determine the performance requirement).

Dimensioning for various parameter settings. As indicated earlier, the required bandwidth should increase when the performance criterion (through ε and T) becomes more stringent. To give a few examples of the impact of the performance
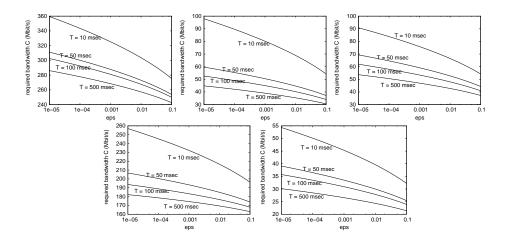
Fig. 4. Required bandwidth for other settings of T and ε for locations $\{U, R, C, A, S\}$, with $M = \{207, 18.9, 23.4, 147, 14.3\}$ Mbit/s, respectively.

parameters T and ε on the required bandwidth capacity, we plot curves for the requires bandwidth capacity at $T = 10, 50, 100$ and 500 msec, and ε ranging from 10^{-5} to 0.1, in Fig. 4. In these curves, M and $V(T)$ are estimated from an example traffic trace collected at each of the locations $\{U, R, C, A, S\}$.

Figure 4 shows that the required bandwidth C decreases in both T and ε, which is intuitively clear. The figures show that C is more sensitive to T than to ε — take for instance the top-left plot in Figure 4, i.e., location U; at $\varepsilon = 10^{-5}$, the difference in required bandwidth between $T = 10$ msec and $T = 100$ msec, is some 20%. At $T = 100$ msec, the difference in required bandwidth between $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-4}$ is just 3% approximately. For other examples, the precise differences may change but the impression stays the same: a tenfold increase in stringency with respect to T requires (relatively) more extra bandwidth, than a tenfold increase in stringency with respect to ε (of course, this could already be expected on the basis of the required link rate formula).

We have verified whether the required link rate is accurately estimated for these case-studies with different settings of T and ε. The estimation errors in these new situations are similar to the earlier obtained results (cf. Table 2). It should be noted however, that we have not been able to verify this for all possible combinations of T and ε: for $\varepsilon = 10^{-5}$ and $T = 500$ msec for instance, there are only 1800 samples in our traffic trace (which has a length of 15 minutes) and hence, we cannot compute the accuracy of our estimation. Another remark that should be made here, is that for locations with only limited aggregation in terms of users (say some tens concurrent users), combined with a small timescale of $T = 10$ msec, the Gaussianity assumption may become questionable. Consequently, the accuracy of our required bandwidth estimation decreases.

Impact of changing performance parameters on required bandwidth. As illustrated in Fig. 4, it is possible to express the estimated required bandwidth capacity as function of ε and T. Having such a function at our disposal, and one or two actual estimates of the required bandwidth, it is possible to 'extrapolate' such

estimates to other settings of ε and T. This allows for investigation of the impact of, say, a more stringent performance target on the required capacity. We first assess the impact of a change in ε and then of a change in T.

Suppose that, for a given T, a proper required bandwidth estimate is known, $C(T, \varepsilon_1)$, for some ε_1 and estimated M. From (2) it follows that $C(T, \varepsilon_1) = M + \delta_1 \cdot \Psi$, where $\delta_1 := \sqrt{-2 \log \varepsilon_1}$. Evidently, we can estimate Ψ by $(C(T, \varepsilon_1) - M)/\delta_1$. Then, to find the required bandwidth estimate for some *other* performance target ε_2, it is a matter of inserting these M and Ψ into

$$C(T, \varepsilon_2) = M + \Psi \sqrt{-2 \log \varepsilon_2}.$$

We give an example application hereof using the top-left graph (location U) in Fig. 4. At the $T = 100$ msec timescale, taking $\varepsilon_1 = 0.01$, $M = 207$ Mbit/s, it follows that $C(T, \varepsilon_1) \approx 266$ Mbit/s. Thus, $\Psi \approx 19.4$. Suppose we are interested in the impact on the required bandwidth capacity if we reduce ε with a factor 1000, i.e., $\varepsilon_2 = 10^{-5}$. Estimating the new required bandwidth capacity through the formula above yields that $C(T, \varepsilon_2) \approx 300$ Mbit/s, which indeed corresponds to the required bandwidth as indicated by the curve in Fig. 4. Hence, informally speaking, the additional bandwidth required to cater for 1000 times as many 'traffic peaks' is, in this scenario, just some 34 Mbit/s.

Secondly, we look at the impact of a change in T on the required bandwidth. Compared to the above analysis for ε, we now have the extra complexity of the variance $V(T)$ in (2), which evidently changes with various T. We therefore impose the additional assumption that traffic can be modeled as fractional Brownian motion (fBm); this special case of the Gaussian model has found widespread use in modeling network traffic. Under fBm, the variance satisfies $V(T) \approx \sigma \cdot T^{2H}$, where H is the so-called Hurst parameter, and σ is some positive scaling constant. Using this variance function, (2) can be rewritten as $C = M + \delta \cdot \Phi(T)$, with $\Phi(T) = \sqrt{\sigma} \cdot T^{H-1}$.

Now suppose that for two different time intervals, namely $T_1 = T$ and $T_2 = \beta T$ (for some $\beta > 0$; ε is held fixed), the required bandwidth is known. This enables us to compute $\Phi(T)$ and $\Phi(\beta T)$, as above. But then

$$\frac{\Phi(\beta T)}{\Phi(T)} = \frac{\sqrt{\sigma} \cdot (\beta T)^{H-1}}{\sqrt{\sigma} \cdot T^{H-1}} = \beta^{H-1},$$

or, in other words, $g := (\log \beta)^{-1} \cdot \log(\Phi(\beta T)/\Phi(T))$ is constant in β (and has value $H - 1$). Again we consider, as an example, location U, with $\varepsilon = 10^{-3}$. For $T = 100$ msec we obtain from $C(T, \varepsilon) \approx 279$ that $\Phi(T) = 19.37$. Now take $\beta = 0.5$; from $C(\beta T, \varepsilon) \approx 290$ we obtain $\Phi(\beta T) = 22.3$ It follows that $g = -0.20$. Suppose we now wish to dimension for $T_3 = \beta' T$ with $\beta' = 0.1$ (i.e., $T = 10$ msec), we obtain $\Phi(\beta' T) = \Phi(T)(\beta')^g \approx 30.7$, so that $C(\beta' T, \varepsilon) = M + \sqrt{-2 \log \varepsilon} \cdot \Phi(\beta' T) \approx 321$. It is easily verified that this corresponds to the required bandwidth as indicated by the curve in Fig. 4.

Dimensioning factors. Link dimensioning formula (2) requires knowledge of M and $V(T)$ to estimate the minimally required link capacity, for specified ε and T.

Location	U			R			C			A			S		
T (sec)	1.0	0.5	0.1	1.0	0.5	0.1	1.0	0.5	0.1	1.0	0.5	0.1	1.0	0.5	0.1
d	1.33	1.35	1.42	2.91	3.12	3.82	1.71	1.83	2.13	1.13	1.14	1.19	1.98	2.10	2.44
σ_d	0.10	0.09	0.09	1.51	1.57	1.84	0.44	0.49	0.67	0.03	0.03	0.03	0.78	0.87	1.01

Table 3. Required bandwidth: dimensioning factors ($\varepsilon = 0.01$).

It is common practice to measure M, for instance through the popular MRTG tool [13]. Operators then look at the 'busy hour' to estimate the load at the busiest time of the day. It is less common to also estimate $V(T)$, which reflects the fluctuations of the traffic rate at the (usually rather small) timescale T — this could be done through the method described in Section 3 of this paper. It would be interesting though to know whether there is a common *dimensioning factor*, say d, which yields the required bandwidth (taking into account fluctuations at small timescales), just on the basis of the mean traffic rate. If there would be such a common dimensioning factor, one could easily estimate the required bandwidth through a simple formula of the type $C = d \cdot M$.

In order to study this dimensioning factor, the required bandwidth and mean traffic rates are compared, by computing $d := C/M$, for each trace at all locations. These dimensioning factors, averaged over all traces at each location, as well as their respective standard deviations, are given in Table 3.

Table 3 shows, for instance, that at location U, some 33% extra bandwidth capacity would be needed on top of the average traffic load M, to cater for 99% ($\varepsilon = 0.01$) of all traffic peaks at a timescale of $T = 1$ sec. At location R, relatively more extra bandwidth is required to meet the same performance criterion: about 191%. Such differences between those locations can be explained by looking at the network environment: at location R, a single user can significantly influence the aggregated traffic, because of the relative low aggregation level (tens of concurrent users) and the high access link speeds (100 Mbit/sec, with a 1 Gbit/sec backbone); at location U, the user aggregation level is much higher, and hence, the traffic aggregate is 'more smooth'. Conclusion is that simplistic dimensioning rules of the type $C = d \cdot M$ are inaccurate, as the d is all but a universal constant (it depends on the nature of the traffic, on the level of aggregation, the network infrastructure, and on the performance target imposed). The table does, however, show, that *within a location* in some situations (in particular locations U and A) the standard deviation of d is rather low; in these cases one could empirically determine d (for fixed T, ε), and dimension through $C = d \cdot M$.

6 Concluding remarks

This paper introduced the concept of 'smart dimensioning'. We derived a dimensioning formula that gives the minimally required bandwidth capacity for a network link. We evaluated this formula using an extensive number of traffic traces collected at different locations. It turned out that the formula accurately predicts the required bandwidth, which is of valuable help when considering link dimensioning as approach to meeting the performance targets agreed upon in the Service Level Agreement.

The main question we posed is that of *how much* additional bandwidth is required, on top of the average rate traffic rate M. From our evaluation, we may conclude that there is no universal multiplicative factor d that would support a statement like 'a bandwidth of $d \cdot M$ suffices'. It is clear that the factor d depends heavily on the performance requirement imposed, but also on the nature of the traffic, the level of aggregation, and the network infrastructure. We have seen that in some scenarios, as low as 13% extra bandwidth (on top of M) is enough, while in others almost this percentage was around 300% (but, evidently, these numbers should be not seen as universal boundaries). Clearly, the '30 times the average traffic rate', as observed by [5] in several real scenarios, seems highly overdone.

References

1. Zhao, W., Olshefski, D., Schulzrinne, H.: Internet Quality of Service: an Overview. Technical report, Columbia University (2000) CUCS-003-00.
2. Pras, A., van de Meent, R., Mandjes, M.: QoS in Hybrid Networks - An Operator's Perspective. In de Meer, H., Bhatti, N.T., eds.: Proceedings of the 13th International Workshop on Quality of Service (IWQoS 2005). Volume 3552 of Lecture Notes in Computer Science (LNCS), Passau, Germany (2005) 388–391
3. Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An Architecture for Differentiated Services. IETF RFC 2475 (1998)
4. Braden, R., Clark, D., Shenker, S.: Integrated Services in the Internet Architecture: An Overview. IETF RFC 1633 (1994)
5. Odlyzko, A.M.: Data networks are lightly utilized, and will stay that way. Review of Network Economics **2** (2003) 210–237
6. Fraleigh, C., Moon, S., Lyles, B., Cotton, C., Khan, M., Moll, D., Rockell, R., Seely, T., Diot, C.: Packet-Level Traffic Measurements from the Sprint IP Backbone. IEEE Network **17** (2003)
7. Fraleigh, C., Tobagi, F., Diot, C.: Provisioning IP Backbone Networks to Support Latency Sensitive Traffic. In: Proceedings of IEEE Infocom, San Francisco, USA (2003)
8. van de Meent, R., Mandjes, M., Pras, A.: Gaussian traffic everywhere? In: Proceedings of the 2006 IEEE International Conference on Communications (ICC 2006), Istanbul, Turkey (2006)
9. van de Meent, R.: Network link dimensioning: a measurement & modeling based approach. PhD thesis, University of Twente (2006)
10. Kilpi, J., Norros, I.: Testing the Gaussian approximation of aggregate traffic. In: Proceedings of the 2nd ACM SIGCOMM Internet Measurement Workshop, Marseille, France (2002) 49–61
11. van den Berg, J., Mandjes, M., van de Meent, R., Pras, A., Roijers, F., Venemans, P.: QoS-aware bandwidth provisioning of IP links. Computer Networks **50** (2006) 631–647
12. Mandjes, M., van de Meent, R.: Inferring traffic characteristics by observing the buffer content distribution. In Boutaba, R., et al., ed.: Proceedings of the 4th International IFIP-TC6 Networking Conference (Networking 2005). Volume 3462 in Lecture Notes in Computer Science (LNCS), Waterloo, Canada (2005) 303–315
13. Oetiker, T.: MRTG: Multi Router Traffic Grapher (2003) Available from URL: http://people.ee.ethz.ch/~oetiker/webtools/mrtg/.