Centrum voor Wiskunde en Informatica

REPORTRAPPORT

PNA

Probability, Networks and Algorithms

Probability, Networks and Algorithms

Network link dimensioning

A. Pras, R. van de Meent, M.R.H. Mandjes,
L.J.M. Nieuwenhuis

REPORT PNA-R0804 APRIL 2008

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3711

Network link dimensioning

ABSTRACT
One of the tasks of network management is to dimension the capacity of access and backbone links. In practice, this type of dimensioning is often based on simple rules of thumb, like `take the maximum bandwidth as measured with MRTG, and add a certain safety margin'. Rules of this type lack preciseness, however, as they fail to reliably predict whether the quality, as agreed upon in the Service Level Agreement, is actually provided. To make better predictions, a more sophisticated mathematical setup is needed. The major contribution of this paper is that it presents such a setup; in this a pivotal role is played by a simple, yet versatile, formula that gives the minimum amount of capacity needed, as a function of the average traffic rate, traffic variance (to be thought of as a measure of `burstiness'), as well as the required performance level. In order to apply the dimensioning formula, accurate estimates of the average traffic rate and the traffic variance are needed. As opposed to the average rate, the traffic variance is rather hard to estimate; this is because measurements on small time scales are needed. We present an easily implementable remedy for this problem, in which the traffic variance is inferred from occupancy statistics of the buffer within the switch or router. To validate the resulting dimensioning procedure, we collected hundreds of traces at multiple (representative) locations, estimated for each of the traces the average traffic rate and (using the approach described above) traffic variance, and inserted these in the dimensioning formula. It turns out that the capacity estimate obtained by the procedure, is usually just a few percent off from the (empirically determined) minimally required value.

2000 Mathematics Subject Classification: 68M10, 90B18
Keywords and Phrases: Link dimensioning, Gaussian models

Dimensioning Network Links

Aiko Pras*, Remco van de Meent†, Michel Mandjes‡, Lambert Nieuwenhuis*
* University of Twente, The Netherlands, {a.pras, l.j.m.nieuwenhuis}@utwente.nl
† Vodafone NL, The Netherlands, Remco.vandeMeent@vodafone.com
‡ University of Amsterdam, The Netherlands, mmandjes@science.uva.nl

***Abstract*— One of the tasks of network management is to dimension the capacity of access and backbone links. In practice, this type of dimensioning is often based on simple rules of thumb, like 'take the maximum bandwidth as measured with MRTG, and add a certain safety margin'. Rules of this type lack preciseness, however, as they fail to reliably predict whether the quality, as agreed upon in the Service Level Agreement, is actually provided. To make better predictions, a more sophisticated mathematical setup is needed. The major contribution of this paper is that it presents such a setup; in this a pivotal role is played by a simple, yet versatile, formula that gives the minimum amount of capacity needed, as a function of the average traffic rate, traffic variance (to be thought of as a measure of 'burstiness'), as well as the required performance level.**

In order to apply the dimensioning formula, accurate estimates of the average traffic rate and the traffic variance are needed. As opposed to the average rate, the traffic variance is rather hard to estimate; this is because measurements on small time scales are needed. We present an easily implementable remedy for this problem, in which the traffic variance is inferred from occupancy statistics of the buffer within the switch or router. To validate the resulting dimensioning procedure, we collected hundreds of traces at multiple (representative) locations, estimated for each of the traces the average traffic rate and (using the approach described above) traffic variance, and inserted these in the dimensioning formula. It turns out that the capacity estimate obtained by the procedure, is usually just a few percent off from the (empirically determined) minimally required value.

I. INTRODUCTION

To ensure that network links are sufficiently provisioned, network managers generally rely on straightforward emprirical rules. They base their decisions on rough estimates of the load imposed on the link, relying on tools like MRTG [1], which poll Management Information Base (MIB) variables like those of the *Interfaces Table*, on a regular basis (for practical reasons, often in five minute intervals). Since the peak load *within* such a measurement interval is in general substantially higher than the average load, one frequently uses rules of thumb like 'take the bandwidth as measured with MRTG, and add a safety margin of 30 percent'.

The problem with such an empirical approach, is that in general it is not obvious how to choose the right safety margin. Clearly, the safety margin is strongly affected by the performance level to be delivered, i.e., that was agreed upon in the Service Level Agreement (SLA); evidently, the stricter the SLA, the higher the capacity needed on top of the average load. Also, traffic fluctuations play an important role here: the burstier the traffic, the larger the safety margin needed. In other words: the simplistic rule mentioned above fails to incorporate the dependence of the required capacity on the SLA and the traffic characteristics. Clearly, it is in the interest of the network manager to avoid inadequate dimensioning. On the one hand, *under*dimensioning leads to congested links, and hence inevitably to performance degradation. On the other hand, *over*dimensioning leads to a waste of capacity (and money); for instance in networks operating under *Differentiated Services* (DiffServ), this 'wasted' capacity could have been used to serve other service classes.

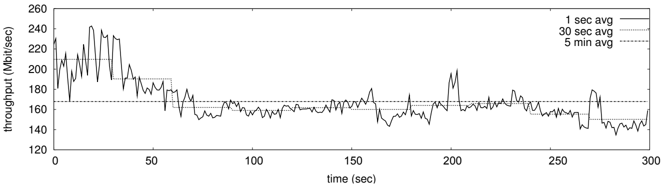

Fig. 1. Traffic rates at different time-scales

We further illustrate this problem by examining one of traces that we have captured. Figure 1 shows a five minute interval of the trace. The five minute traffic average throughput is around 170 Mbps. The traffic average throughput of the first thirty seconds period equals around 210 Mbps, i.e., 30% higher than the five minute average. Some of the one second traffic average throughput values go up to 240 Mpbs, i.e., more than 40% of the five minute average values. Although not shown in the figure, we even measured ten millisecond spikes of more than 300 Mbps, which is almost twice as much as the five minutes value. Hence, the average traffic throughput strongly depends on the time-period over which the average is determined. We therefore conclude that rules of thumb like 'take the maximum bandwidth as measured with MRTG, and add a safety margin of 30 percent' lack general validity, and are therefore oversimplistic, in that they give inaccurate estimates of the amount of capacity needed.

The above reasoning stresses the need for a more generic setup that encompasses the traffic characteristic (such as average traffic rate, and some measure for burstiness or traffic variance), the performance level to be achieved, and the required capacity. Qualitatively, it is clear that more capacity is needed if the traffic supply increases (both in terms of the rate and the burstiness) or the performance requirements are more stringent, but in order to successfully dimension network links, one should have *quantitative* insights into these interrelationships as well.

The goal of this paper is to develop a methodology that can be used for determining the capacity that is needed on Internet links, given specific performance requirements. Our methodoly is based on a dimensioning formula, that describes the above-mentioned trade-offs between traffic, performance, and capacity. In our approach, the traffic profile is summarized by the average traffic rate and the traffic variance (to be thought of as a measure of burstiness). Given predefined performance requirements, we are then in a position to determine the required capacity of the network link, by using estimates of the traffic rate and traffic variance.

We argue that particularly the traffic variance is not straightforward to estimate, especially on smaller time scales as mentioned above. We circumvent this problem by relying on an advanced estimation procedure based on occupancy statistics of the buffer within the switch or router, so that, importantly, it is *not* necessary to measure traffic at these small time scales. We extensively validated our dimensioning procedure, using hundreds of traffic traces that we collected at various locations, and which differ substantially, both in terms of size and in the types of users. For each of the traces we estimated the average traffic rate and traffic variance, using the above mentioned buffer-occupancy method. At the same time, we empirically determined per trace also the correct capacities, that is, the mimimum capacity needed to satisfy the performance requirements. Our experiments indicate that the determined capacity of the needed internet link is highly accurate, and usually just a few percent off from the correct value.

The material presented in this paper was part of a larger project, that culminated in the thesis [2]; in fact, the idea behind this paper is to present the main results of that study to a broad audience. Mathematical equations are therefore kept to a minimum. Readers interested in the mathematical background or other details are therefore referred to the thesis [2] and other publications [3], [4].

The structure of this paper is as follows. Section II presents the dimensioning formula that yields the capacity needed to provision an Internet link, as a function of the traffic characteristics and the performance level to be achieved. Section III discusses how this formula can be used in practice; particular attention is paid to the estimation of the traffic characteristics. To assess the performance of our procedure, Section IV compares the capacity estimates with the 'correct' values, using hundreds of traces.

II. DIMENSIONING FORMULA

An obvious prerequisite for a dimensioning procedure is a precisely defined performance criterion. It is clear that a variety of possible criteria can be chosen, with their specific advantages and disadvantages. We have chosen to use a rather generic performance criterion, which we refer to as *link transparency*. Link transparency is parametrized by two parameters, viz. a time interval T and a fraction ε, and is defined as: *the fraction of disjoint (time-)intervals of length T in which the offered traffic exceeds the link capacity C, should be below ε.*

The link capacity required under link transparency, say $C(T, \varepsilon)$, depends on the parameters T, ε, but clearly also on the characteristics of the offered traffic. If we take, for example, ε = 1%, and T = 100 ms, our criterion says that in no more than 1% of the time intervals of length 100 ms the offered load is supposed to exceed the link capacity C. T represents the time interval over which the offered load is measured; for interactive applications like web browsing this interval should be short, say in the range of tens or hundreds of milliseconds up to 1 second. It is intuitively clear that a shorter time interval T and/or a smaller fraction ε, will lead to a higher required capacity C. We note that the choice of suitable values for T and ε is primarily the task of the network operator; he should choose a value that suits his (business) needs best. It is clear that the specific values evidently depend on the underlying applications, and should reflect the Service Level Agreements agreed upon with the end-users.

Having introduced our performance criterion, we now proceed with presenting a (quantitative) relation between traffic characteristics, the desired performance level, and the link capacity needed. In earlier papers we have derived (and thoroughly studied) the following formula to estimate the minimum required capacity of an Internet link [2], [3]:

$$C(T, \varepsilon) = \mu + \frac{1}{T}\sqrt{(-2\log\varepsilon)\cdot v(T)}; \tag{1}$$

this dimensioning formula shows that the required link capacity $C(T, \varepsilon)$ can be estimated by adding to the average traffic rate μ some kind of 'safety margin'. Importantly, however, in contrast to equating it to a fixed number, we give an explicit an insightful expression for it: we can determine the safety margin, given the specific value of the performance target and the traffic characteristics.

In the first place it depends on ε through the quare root of it natural logarithm — it, for instance, says that replacing $\varepsilon = 10^{-4}$ by $\varepsilon = 10^{-7}$ means that the safety margin has to be increased by about 32% (use that $\sqrt{\log 10^{-7}/\log 10^{-4}}$ equals $\sqrt{7/4} = 1.32$). Secondly, it depends on time interval T. The parameter $v(T)$ is called the *traffic variance*, and represents the variance of traffic arriving in intervals of length T. The traffic variance $v(T)$ can be interpreted as a kind of burstiness and is typically (roughly) of the form αT^{2H}, for $H \in (\frac{1}{2}, 1)$, $\alpha > 0$ [6], [7]. We see that the capacity needed on top of μ is proportional to T^{H-1} and, hence, increases when T decreases, as could be expected, cf. the example trace corresponding to Fig. 1. In the third place, the required capacity obviously depends on the traffic characteristics, both through the 'first order estimate' μ and the 'second order estimate' $v(T)$. We emphasize that safety margins should not be thought of as fixed numbers, like the 30% mentioned in the introduction; instead, it depends on the traffic characteristics (i.e., it increases with the burstiness of the traffic), as well as on the strictness of the performance criterion imposed.

It is important to realize that our dimensioning formula assumes that the underlying traffic stream is *Gaussian*. In our research we therefore extensively investigated whether this assumption holds in practice; due to central-limit-theorem type of arguments, one expects that it should be accurate as long as the aggregation level is sufficiently high. We empirically found that aggregates resulting from just a few tens of users already make the resulting traffic stream fairly Gaussian; see [5] for precise statistical support for this claim. In many practical situations one can therefore safely assume Gaussianity; this conclusion is in line with what is found elsewhere [6]–[8].

III. HOW TO USE THE DIMENSIONING FORMULA

The dimensioning formula presented in Section II requires four parameters: ε, T, μ and $v(T)$. As argued above, the performance parameter ε and the time interval T must be chosen by the network manager and can, in some cases, directly be derived from a Service Level Agreement. Possible values for these parameters are $\varepsilon = 1\%$ (meaning that the link capacity should be sufficient in 99% of the cases) and T = 100 ms (popularly speaking, in the exceptional case that the link capacity is not sufficient, the overload situation does not last longer than 100 ms). The two other parameters, the average traffic rate μ and the traffic variance $v(T)$, are typically less straightforward to determine, and their estimation will therefore be discussed in separate subsections below.

A. Example

Before discussing the various ways to determine the average traffic rate and variance, a short example of a university backbone link will be presented first. In this example we have chosen $\varepsilon = 1\%$ and $T = 100$ ms. To find μ and $v(T)$, we have measured all traffic flowing over the university link for a period of 15 minutes. From this measurement we have measured the average traffic rate for each 100 ms interval within these 15 minutes; this rate is shown as the plotted line in Figure 2. The figure indicates that this rate varies between 125 and 325 Mbps. We also measured the average rate μ over the entire 15 minutes interval ($\mu = 239$ Mbps), as well as the traffic variance over intervals of length $T = 100$ ms, i.e., $v(T)$ ($\sqrt{v(T)} = 2.7$ Mbit; note that the square root of the variance is the standard deviation, which is more straightforward to interpret). After inserting the four parameter values into our formula, we found that the required capacity for the university access link should be $C = 320.8$ Mbps. This capacity is drawn as straight line in the figure. As can be seen, this capacity is sufficient for most of the time; we empirically checked that this was indeed the case in about 99% of the 100 ms intervals.

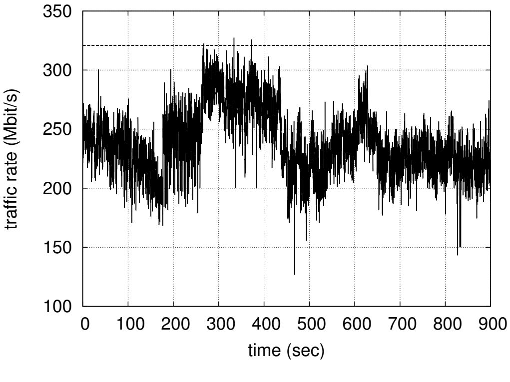

Fig. 2. Example from a university access link

B. Approaches to determine the average traffic rate

The average traffic rate μ can be estimated by measuring the amount of traffic (the number of bits) crossing the Internet link, which should then be divided by the length of the measurement window (in seconds). For this purpose the manager can connect a measurement system to that link and use tools like `tcpdump`. To capture usage peaks, the measurement could run for a longer period of time, for example a week. If the busy period is known, for example each morning between 9:00 and 9:15, it is also possible to measure during that period only.

The main drawback of this approach is that a dedicated measurement system is needed. The system must be connected to the network link, and be able to capture traffic at line speed. At Gigabit speed and faster this may be a highly non-trivial task. Fortunately the average traffic rate μ can also be determined by using SNMP and read the `ifHCInOctets` and `ifHCOutOctets` counters from the Interfaces MIB. This MIB is implemented in most routers and switches, although old equipment may only support the 32 bit variants of these counters. Since 32 bit counters may wrap within a measurement interval, it might be necessary to poll the values of these counters on a regular basis; if 64 bit counters are implemented, it is sufficient to retrieve the values only at the begin and end of the measurement period. Anyway, the total number of transferred bits as well as the average traffic rate can be determined by performing some simple calculations. Compared to using `tcpdump` at Gigabit speed, the alternative of using SNMP to read some MIB counters is rather attractive, certainly in cases where operators already use tools like MRTG [1], which perform these calculations automatically.

C. Direct approach to determine traffic variance

Like the average traffic rate μ, also the traffic variance $v(T)$ can be determined by using `tcpdump` and directly measure the traffic that is flowing over the Internet link. To determine the variance, however, it is now *not* sufficient to know the total amount of traffic exchanged during the measurement period (of 15 minutes); instead, it is necessary to measure the amount of traffic for every interval of length T, i.e., in our example 1500 measurements at 100 ms intervals. This will result in a series of traffic rate values; the traffic variance $v(T)$ can then be estimated in a straightforward way from these values by applying the standard sample variance estimator.

It should be noted that, as opposed to the average traffic rate μ, now it is *not* possible to use the `ifHCInOctets` and `ifHCOutOctets` counters from the Interfaces MIB. This is because the values of these counters must now be retrieved after every interval T, thus, in our example, after every 100 ms. Fluctuations in SNMP delay times [10], however, are such that it will be impossible to obtain the precision that is needed for our goal of link dimensioning. At first sight it therefore seems unavoidable to still connect a measurement system to the link and use tools like `tcpdump` to determine traffic variance. Since this is unattractive, the next subsections discuss the use of alternative MIB variables, which need not be polled on short intervals like T, and thus will not be affected by fluctuations in SNMP delay times.

D. Indirect approach to determine traffic variance

One of the major outcomes of our research [2] is an *indirect* procedure to estimate the traffic variance, having the attractive property that it avoids measurements on small time scales. This indirect approach exploits the relationship that exists between $v(T)$ and the occupancy of the buffer (in the router or switch) in front of the link to be dimensioned. This relationship can be expressed through the following formula [2]: for any t,

$$v(t) \approx \min_{B>0} \frac{(B + (C_q - \mu)t)^2}{-2 \log \mathbb{P}(Q > B)}. \tag{2}$$

In this formula, C_q represents the current capacity of the link, μ the average traffic rate over that link, and $\mathbb{P}(Q > B)$ the buffer content's (complementary) distribution function (that is, the fraction of time the buffer level Q is above B). The formula shows that, once we know the buffer contents distribution $\mathbb{P}(Q > B)$, we can for any t, study

$$\frac{(B + (C_q - \mu)t)^2}{-2 \log \mathbb{P}(Q > B)} \tag{3}$$

as a function of B, and its minimal value provides us with an estimate of $v(t)$. In this way, we can *infer* $v(t)$ for any time-scale t — by choosing $t = T$, we indeed find an estimate of $v(T)$, which was needed in our dimensioning formula. Theoretical justification of (2) can be found in [11].

The remaining question is now how $\mathbb{P}(Q > B)$ can be estimated. To answer that question, let us assume that a MIB variable exists that represents the amount of data in the buffer located in front of the link. This MIB variable should be read multiple times to collect N 'snapshots' of the buffer contents $q_1, \ldots, q_N$. Obviously, from these snapshots we are now able to estimate the buffer contents distribution $\mathbb{P}(Q > B)$. To determine $v(t)$, we have to fill in each possible value of B in the above formula, with $t = T$, and find that specific B for which (3) is minimal; this minimal value is then the estimate of the traffic variance we are looking for.

The advantage of this indirect approach is that it is no longer necessary to measure traffic at time scale T to determine $v(T)$. Instead, with this indirect approach, it is sufficient to take a number of snapshots from a MIB variable representing the occupancy of the buffer in front of the link. In our research we proved that the intervals between the various snapshots can be taken fairly randomly — there is no need to take equally sized intervals, which is an important advantage of the indirect procedure. Further results on the number of buffer snapshots needed to obtain a reliable estimate of $\mathbb{P}(Q > B)$, and on the measurement frequency, are presented in detail in [2].

E. Implementation requirements for the indirect approach

The indirect approach requires the existence of a MIB variable representing the length of the output queue, but such a variable has not been standardized by the IETF yet. The variable that comes closest is `ifOutQLen` from the Interfaces MIB. In the latest specifications of this MIB module the status of this variable has been deprecated, however, which means that this variable is obsolete, although implementors may still implement it to ensure backwards compatibility. In addition, the `ifOutQLen` variable measures the length of the queue in packets, whereas our procedure requires the queue length to be in bits. Although this 'incompatibility' might be 'fixed' by means of some probabilistic computations, our recommendation is to add to the definition of some MIB module a variable representing the length of the output queue in bits (or octets). We stress that implementing such variable should be straightforward; Random Early Detection (RED) queueing algorithms, which are widely implemented in modern routers, already keep track of this information.

A second issue regarding the indirect approach is that it may seem impossible to estimate a 'usable' buffer content distribution $\mathbb{P}(Q > B)$. For example, if the capacity of the outgoing link is much higher than the traffic rate, the buffer in front of that link will (nearly) always be empty. Also in case the traffic rate approaches the link capacity, the buffer in front of that link becomes overloaded, so that we do not have any useful information on the buffer content distribution for small values of B. To circumvent these complications, vendors of switches and routers could implement some kind of 'intelligence' within their devices. Such intelligence could simulate the queueing dynamics of a *virtual* queue, with a virtual outgoing line with capacity C_q, that can be chosen smaller or larger than the actual capacity. If the link is underloaded, the capacity of the virtual queue should clearly be chosen substantially smaller than the actual capacity, in order to obtain an informative estimate of the buffer content distribution; if the link is overloaded, then vice versa. Procedures for detecting appropriate values for the virtual capacity are presented in [2]. Figure 3 shows the structure of such intelligence within a switch or router. Since RED-enabled routers already include much of this intelligence, implementation will be relatively straightforward.

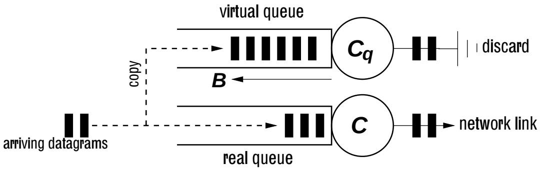

Fig. 3. Decoupling the real queue from a virtual queue

IV. VALIDATION

In this section the correctness of our link dimensioning procedure will be validated in two steps. For each trace:

- First, we validate the correctness of formula (2). We do this by comparing the results of the direct approach to determine traffic variance (see Subsection III-C) to the results obtained via the indirect approach that was based on formula (2) (see Subsection III-D).
- Second, we validate the correctness of formula (1). We empirically determine the 'correct' value of the link capacity, that is, we empirically find the minimum service rate needed to meet the performance criterion (T, ε). We then compare the outcome of formula (1) with this 'correct' capacity.

Subsection IV-A starts with providing details about the measurements that were needed to perform the validation. Subsection IV-B presents the comparison between the direct and indirect approach. Then finally Subsection IV-C compares the outcome of formula (1) with the empirical approach.

A. Measurements

To enable a thorough validation study, we have collected around 850 TCP/IP packet traces, based on measurements performed between 2002 and 2006. To ensure that the traffic within these traces is representative for large parts of the Internet, we have measured on five different types of links:

- *A*: A 1 Gbit/s uplink of an ADSL access network. Several hundreds of ADSL customers are connected to this network; the link capacity for each individual ADSL user varies between 256 Kbit/s and 8 Mbps.
- *C*: A 1 Gbit/s link between a large college network and the Dutch academic and research network (SURFnet). This college network serves around 1000 students, most of them being connected via 100 Mbps Ethernet links.
- *R*: A 1 Gbit/s link between a research institute and SURFnet. The research network is used by approximately 200 researchers, each having a 100 Mbps link to the research network.
- *S*: A 50 Mbps Internet access link of a server-hosting company. This company provides floor- and rack-space to clients who want to connect, for example, their web-servers to the Internet. Internally, most servers are connected via 100 Mbps links.
- *U*: A 300 Mbps (three parallel 100 Mbps Ethernet links) between the residential and the core network of a university. Around 2000 students are each connected via 100 Mbps links to this residential network; an important share of the traffic generated by these students remains within this residential network and is therefore not visible on the link towards the university's core network.

Each trace contains 15 minutes worth of TCP/IP header data; the sizes of these traces range from a few Megabytes to a few Gigabytes. In total some 500 Gigabytes of TCP/IP header data was collected. This data has been anonymized and can be downloaded from our web server [9].

B. Traffic variance: direct versus indirect approach

In this subsection we compare the traffic variance as can be estimated from direct link measurements (the direct approach) to the traffic variance that can be estimated using formula (2), i.e., the approach that measures the occupancy distribution of the buffer in front of the link (the indirect approach), with an appropriately chosen value of the virtual queue's link capacity.

Since MIB variables that represent the buffer occupancy within routers and switches are not yet available, we chose to simulate such a router. The simulator consists of a number of `Perl` scripts, which implement a virtual queue; this queue is similar to the queue shown in Figure 3. Input to the scripts were the packet traces discussed in the previous subsection. Since these traces include timestamps indicating the precise moment each packet was captured, replaying the traces was straightforward. After a new packet was offered to the simulator, the resulting buffer occupancy could be estimated by adding the length of this packet to the length of the current queue. In addition, the simulator could estimate the speed at which the queue was emptied using the capacity C_q of the outgoing link. To determine the buffer contents distribution $\mathbb{P}(Q > B)$, the simulator included a parameter that represented the current buffer occupancy. This parameter was read many times, to ensure that we have a sufficient number N of snapshots to reliably estimate $\mathbb{P}(Q > B)$. We also estimated for every trace the average traffic rate μ. In this way we obtained all information needed in formula (2).

trace	$\sqrt{v_{\mathsf{direct}}(T)}$	$\sqrt{v_{\mathsf{indirect}}(T)}$	μ
loc. *A* - 1	0.969	1.032	147.180
loc. *A* - 2	0.863	0.864	147.984
loc. *C* - 1	0.796	0.802	23.894
loc. *C* - 2	3.263	3.518	162.404
loc. *R* - 1	0.701	0.695	18.927
loc. *R* - 2	0.241	0.249	3.253
loc. *S* - 1	0.447	0.448	14.254
loc. *S* - 2	0.152	0.152	2.890
loc. *U* - 1	1.942	2.006	207.494
loc. *U* - 2	2.704	2.773	238.773

TABLE I

DIRECT VERSUS INDIRECT APPROACH

Table I shows, for each of the five locations, the results for two representative traces. It shows, in Mbit, the square root of the traffic variance $v(T)$, thus the standard deviation, for the direct as well as the indirect approach. We note that the table also shows the average traffic rate μ, which is in Mbps. To support real-time interactive applications, the time interval T of our performance criterion was chosen to be 100 ms.

The table shows that there is just a modest difference between the traffic variance obtained using formula (2), and the one obtained using direct link measurements. In many cases the results using formula (2) differ only a few percent from the direct results. The worst result is obtained for loc. C ex. #2; in this case the difference is about 16 percent. Observe, however, that this table may give an overly pessimistic impression, as the dimensioning formula (1) indicates that the error made in the estimation of the capacity is substantially smaller: on the basis of the direct variance estimate (with $\varepsilon = 1\%$) the capacity is estimated to be 261.4 Mbps, and on the basis of the indirect variance estimate 269.2 Mbps, a difference of just 3%.

For space reasons, Table I shows only the results for some traces, but the same kind of results have been obtained for the other traces; see for an extensive set of experiments [2]. Also, results did not change significantly when we selected other values for the time interval T. We therefore conclude that our indirect approach is sufficiently accurate. This also means that in principle there is no need for line-speed measurements to determine traffic variance. Our experiments show that simple MIB variables indicating current buffer occupancy are sufficient for that purpose.

C. Required link capacity

Finally this subsection validates the correctness of formula (1), and thus our approach to dimension network links. This is done by comparing the outcomes of three different approaches:

- Approach A: in this approach we have measured all traffic that is flowing over a certain link, and empirically determined the mimimum capacity needed to meet the performance criterion; this capacity could be considered as the 'correct' value. Although it is difficult to perform such measurements at Gigabit speed and higher, the estimation of the minimum capacity needed to satisfy our performance criterion is rather straightforward (assuming that the link is not yet overloaded).
- Approach B: in this approach we have used formula (1) to determine the required link capacity. The average traffic rate μ as well as the traffic variance $v(t)$ have been determined in the way as described in Subsections III-B and III-C, i.e., the variance has been estimated through the direct procedure.
- Approach C: in this approach we have used both formula (1) and (2). Compared to approach B, the traffic variance $v(t)$ has now been derived from the occupancy of the buffer that is in front of the link, as described in Subsection III-D, i.e., through the indirect procedure.

For all three approaches we have used the same performance criterion: the link capacity should be sufficient in 99% of the cases (ε = 1%) and, in the exceptional case that the link capacity is not sufficient, the overload situation should not last longer than 100 ms (T = 100 ms). Note that results using other performance criteria can be found in [2]; the findings agree to a large extent with those presented here.

Table II shows the outcome for the three approaches, using the same traces as before. The column C_{A} shows, in Mbps, the minimally required link capacity to meet the performance criterion, that we (empirically) found after measuring all traffic flowing over that link. In fact, this is the actual capacity that would be needed in practice to satisfy our performance criterion; it is therefore our target value. Column C_{B} shows the capacity that has been estimated using formula (1); column C_{C} shows the capacity that has been estimated if additionally formula (2) has been used to determine the traffic variance. As shown in the last two columns, the estimated values divided by the target values are very close to 1; in all cases the differences are less than 7 percent.

trace	C_{A}	C_{B}	C_{C}	$\varDelta_{\mathsf{B/A}}$	$\varDelta_{\mathsf{C/A}}$
loc. *A* - 1	171.191	176.588	178.480	1.032	1.043
loc. *A* - 2	168.005	174.178	174.218	1.037	1.037
loc. *C* - 1	44.784	48.033	48.250	1.073	1.077
loc. *C* - 2	265.087	261.444	269.182	0.986	1.015
loc. *R* - 1	37.653	40.221	40.020	1.068	1.063
loc. *R* - 2	10.452	10.568	10.793	1.011	1.033
loc. *S* - 1	27.894	27.843	27.873	0.998	0.999
loc. *S* - 2	7.674	7.482	7.532	0.975	0.981
loc. *U* - 1	258.398	266.440	268.385	1.031	1.039
loc. *U* - 2	302.663	320.842	322.934	1.060	1.067

TABLE II

LINK CAPACITY FOR EACH OF THE THREE APPROACHES

Our procedure to determine link capacity has not only been validated for the 10 traces shown in Table II, but for all 850 traces that were collected as part of our studies. The overall results for the complete procedure, thus approach C, is shown in column 2 and 3 (avg $\varDelta_{\mathsf{C/A}}$ and stderr $\varDelta_{\mathsf{C/A}}$) of Table III. For all locations but *R*, $\varDelta_{\mathsf{C/A}}$ is very close to 1, indicating that the bandwidth as estimated through our procedure is nearly correct.

traces	avg $\varDelta_{\mathsf{C/A}}$	stderr $\varDelta_{\mathsf{C/A}}$	avg $\varDelta_{\mathsf{C/A}}$ *	stderr $\varDelta_{\mathsf{C/A}}$ *
loc. *A*	1.04	0.02	1.04	0.01
loc. *C*	1.04	0.11	1.05	0.08
loc. *R*	0.90	0.19	1.00	0.10
loc. *S*	0.99	0.10	1.01	0.05
loc. *U*	1.01	0.07	1.03	0.06

TABLE III

OVERALL VALIDATION RESULTS

The deviation at location *R* is caused by the fact that at *R* traffic is on average 'less Gaussian' compared to the other measurement locations — as our methodology assumes Gaussian traffic, some error in the resulting estimate can be expected when the traffic is 'not so Gaussian'. To further investigate this, we recomputed all values, but removed the traces that were 'less Gaussian'. Column 4 and 5 of Table III show the results; the differences are now 5% or less. It should be noted that in all cases this difference results in a slight over-estimation of the required capacity; in practice this may be desirable, in particular if meeting the SLA is valued more than (temporarily) not using all transmission capacity available.

V. CONCLUSIONS

Motivated by the fact that rules of thumb usually lead to unreliable capacity estimates, this paper focused on the development of a generic methodology for link dimensioning. It was demonstrated that the capacity of Internet links can be accurately estimated using a simple formula, which requires only four parameters (Section II). The first two of these parameters reflect the desired performance level (representing how often the offered load may exceed the available capacity, and for how long this link exceedance may last), and should be chosen by the network manager. The last two parameters reflect the characteristics of the offered traffic, and can be obtained by estimating the average link load and variance. The average link load can be easily determined by reading certain MIB variables via SNMP; tools like MRTG can be used for that purpose (Section III-B). Measuring traffic variance is somewhat more involved, but may be performed in a sophisticated, indirect way, using the distribution of the occupancy of the buffers located (in the router or switch) in front of the link to be dimensioned (Section III-D). The advantage of this indirect approach is that measurements at small time scales (whose reliability cannot be guaranteed) are no longer needed. Although much of the intelligence to determine the buffer occupancy distribution is already implemented in current routers, the corresponding MIB variables are not yet available. Implementing these variables is argued to be straightforward, however (Section III-E). Our formula has been validated using 850 TCP/IP traces, collected at five different locations, ranging from ADSL access networks, university networks, college networks as well as access links to server hosting companies and research institutes. The validation showed that our formula was able to determine the required link capacity with an error margin of just a few percent (Section IV); our approach therefore clearly outperforms the simple rules of thumb that are usually relied on in practice.

VI. ACKNOWLEDGMENTS

The work reported in this paper was supported in part by the EC IST-EMANICS Network of Excellence (#26854).

REFERENCES

[1] T. Oetiker, “MRTG: Multi Router Traffic Grapher,” 2003, http://people.ee.ethz.ch/~oetiker/webtools/mrtg/.

[2] R. van de Meent, “Network Link Dimensioning - A Measurement and Modelling based approach”, Ph.D. thesis, University of Twente, 2006, http://purl.org/utwente/56434.

[3] J.L. van den Berg, M.R.H. Mandjes, R. van de Meent, A. Pras, F. Roijers, and P.H.A. Venemans, “QoS aware bandwidth provisioning of IP links”, in *Computer Networks*, Vol. 50, No. 5, 2006.

[4] C. Fraleigh, “Provisioning Internet Backbone Networks to Support Latency Sensitive Applications”, Ph.D. thesis, Stanford University, 2002.

[5] R. van de Meent, M.R.H. Mandjes and A. Pras, “Gaussian traffic everywhere?”, in *Proceedings of the 2006 IEEE International Conference on Communications (ICC 2006)*, Istanbul, Turkey, 2006.

[6] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and C. Diot, “Packet-Level Traffic Measurements from the Sprint IP Backbone”, in *IEEE Network*, Vol. 17, No. 6, 2003.

[7] J. Kilpi and I. Norros, “Testing the Gaussian approximation of aggregate traffic”, in *Proceedings of the 2nd ACM SIGCOMM Internet Measurement Workshop*, pages 49-61, Marseille, France, 2002.

[8] R. Guérin and H. Ahmadi and M. Naghsineh, “Equivalent Capacity and its Application to Bandwidth Allocation in High-Speed Networks”, in *IEEE Journal on Selected Areas in Communication*s, Vol. 9, No. 7, 1991.

[9] R. van de Meent and A. Pras, “Traffic measurement repository”, 2007, http://traces.simpleweb.org/.

[10] A. Pras, T. Drevers, R. van de Meent and D. Quartel, “Comparing the Performance of SNMP and Web Services-Based Management”, in *IEEE electronic Transactions on Network and Service Management*, Vol. 1, No. 2, 2004.

[11] M. Mandjes, “A note on the benefits of buffering”, in *Stochastic Models*, Vol. 20, No. 1, 2004.