UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Resource dimensioning through buffer sampling

Mandjes, M.; van de Meent, R.

DOI
10.1109/TNET.2008.2009989
Publication date
2009
Document Version
Final published version
Published in
IEEE/ACM Transactions on Networking

Link to publication

Citation for published version (APA):
Mandjes, M., & van de Meent, R. (2009). Resource dimensioning through buffer sampling. *IEEE/ACM Transactions on Networking*, *17*(5), 1631-1644. https://doi.org/10.1109/TNET.2008.2009989

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:15 Sept 2026

Resource Dimensioning Through Buffer Sampling

Michel Mandjes and Remco van de Meent

***Abstract*—Link dimensioning, i.e., selecting a (minimal) link capacity such that the users' performance requirements are met, is a crucial component of network design. It requires insight into the interrelationship among the traffic offered (in terms of the mean offered load M, but also its fluctuation around the mean, i.e., 'burstiness'), the envisioned performance level, and the capacity needed. We first derive, for different performance criteria, theoretical dimensioning formulas that estimate the required capacity c as a function of the input traffic and the performance target. For the special case of Gaussian input traffic, these formulas reduce to $c = M + \alpha V$, where α directly relates to the performance requirement (as agreed upon in a service level agreement) and V reflects the burstiness (at the timescale of interest). We also observe that Gaussianity applies for virtually all realistic scenarios; notably, already for a relatively low aggregation level, the Gaussianity assumption is justified.**

As estimating M is relatively straightforward, the remaining open issue concerns the estimation of V. We argue that particularly if V corresponds to small time-scales, it may be inaccurate to estimate it directly from the traffic traces. Therefore, we propose an indirect method that samples the buffer content, estimates the buffer content distribution, and 'inverts' this to the variance. We validate the inversion through extensive numerical experiments (using a sizeable collection of traffic traces from various representative locations); the resulting estimate of V is then inserted in the dimensioning formula. These experiments show that both the inversion and the dimensioning formula are remarkably accurate.

***Index Terms*—Buffer sampling, Gaussian traffic, inversion, large deviations, network dimensioning, quality-of-service.**

I. Introduction

ADEQUATE resource dimensioning requires a thorough insight into the interrelationship among: 1) the traffic offered (in terms of the average load, but also its fluctuations), 2) the desired level of performance, and 3) the required capacity. It is clear that more capacity is needed when the offered load becomes higher, the fluctuations become fiercer, or the performance criterion becomes more stringent. However, to make precise predictions about the *amount* of capacity that should be added, advanced modeling and performance techniques are required. These predictions are of crucial importance, as scarce dimensioning inevitably leads to performance degradation, whereas "generous" dimensioning policies essentially result in a waste of resources.

Manuscript received September 11, 2006; revised August 08, 2007 and May 27, 2008; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor M. Roughan. First published June 30,2009; current version published October 14, 2009.

M. Mandjes is with the University of Amsterdam, 1090 GB Amsterdam, The Netherlands (e-mail: mmandjes@science.uva.nl).

R. van de Meent is with the University of Twente, 7500 AE Enschede, The Netherlands and also with Vodafone NL, 6229 GK Maastricht, The Netherlands (e-mail: remco@vandemeent.net).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2008.2009989

In the context of dimensioning IP nodes, several additional considerations play a role. IP networks usually carry a broad variety of traffic types, of which some can tolerate relatively substantial latency (e-mail, particular Web applications), whereas others have strict delay requirements (interactive services). As argued by Fraleigh *et al.* [14], two approaches can be followed: 1) actively discriminating by traffic differentiation mechanisms (i.e., preferential treatment for the more demanding traffic types), or 2) sufficiently provisioning resources so that all traffic types meet their performance requirements (i.e., without any traffic differentiation). In the former option, the cost associated to the management and operation of the network is relatively high, while the efficiency gain (compared to the latter option) is usually modest (particularly when the level of aggregation is high). Therefore, following [14], the focus in this paper is on dimensioning as the approach for delivering performance requirements. (As an aside, we mention that even if one opts for applying traffic differentiation, there is often still a need for dimensioning as well. For instance, as argued in [5], if *diffserv* is applied, then bandwidth dimensioning for traffic aggregates still needs to be done. Put differently, the distinction between traffic differentiation and resource dimensioning is not sharp).

We identified three crucial prerequisites for link dimensioning: 1) a reliable traffic model, 2) a performance target to be met, and 3) formulas computing the performance for a given traffic model and given network resources. Having these at our disposal, we can find expressions for the minimum link capacity required in order to offer a traffic stream with given characteristics a given performance target; we refer to these as *dimensioning formulas*. In Section II, we present a number of generic dimensioning formulas—generic in the sense that they are valid for any (stationary) traffic stream. These formulas, however, are still quite implicit; they require knowledge of the full moment generating function (mgf) of the traffic offered at any time-scale, something that is typically hard to measure and estimate. To overcome this problem it would be helpful if we could restrict ourselves to specific classes of, still sufficiently general and versatile, random processes.

Two such classes of traffic models have been discussed extensively in the literature. The first is what could be called "flow-oriented," where *flows* could be TCP connections, UDP streams, etc. In this approach, flows are modeled (arrival rate, duration, traffic rate while being active, etc.); it is noted that tools like NetFlow (in Cisco routers) already allow for measurement of some flow characteristics (e.g., size and duration). Several papers have found, in different situations, accurate flow-oriented traffic models; see, for instance, [2] and [3]. An alternative is to model the *aggregate* stream rather than individual flows. *Gaussian models* (of which fractional Brownian motion is a special case) form a class of models that is specifically suitable for describing highly aggregated streams. The Gaussian model consists of 1) a *mean rate* M (to which we also refer as "load") and 2) a *variance curve* $V(\cdot)$ (so that $V(t)$

1063-6692/$26.00 © 2009 IEEE

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

corresponds to the variance of the amount of traffic offered in an arbitrary time window of length $t \geq 0$). Having observed that the dimensioning formulas of Section II greatly simplify for Gaussian traffic, in Section III we further assess in detail to what extent one can assume Gaussianity. We discuss which factors affect the Gaussianity (the length of the measurement interval, the level of aggregation) and conclude that in virtually any representative situation (where we considered various locations and points in time) the Gaussian model fits well. We emphasize that, particularly under high load, flow-oriented models and aggregate-stream models do not exclude one another; we come back to this issue later.

Assuming Gaussianity, the dimensioning formulas of Section II require estimates of the mean traffic rate M and the variance curve $V(\cdot)$ to find the required bandwidth for a given performance target. Estimating M is relatively straightforward and can be done through rough traffic measurements (for instance, over 5-min intervals). Estimating the variance curve, however, could be substantially harder. Particularly on smaller time-scales, it is hard to do accurate measurements through simple network management protocol (SNMP). Section IV presents a novel, efficient technique for estimating $V(\cdot)$ by coarse-grained sampling the buffer content, estimating the buffer content distribution, and "inverting" this into the variance curve. Importantly, this procedure eliminates the need for traffic measurements on small time-scales; instead, we measure (for instance at a constant frequency, but this is by no means necessary) the buffer content. In this sense, we remark that the procedure we propose is rather counterintuitive. One would expect that one needs measurements of the traffic offered in intervals of length T to accurately estimate $V(T)$, but apparently one can alternatively sample the buffer content. In fact, one of the attractive features of our 'inversion approach' is that it yields the *entire* variance curve $V(\cdot)$ (evidently up to some finite-time horizon), rather than just $V(T)$ for some prespecified T.

Section V assesses the accuracy of the inversion approach through simulation experiments with both synthetic traffic and real network traces. These validations show excellent performance in the sense that the $V(\cdot)$-curve is indeed estimated well, even if the traffic stream deviates considerably from Gaussian. We also investigate the required measurement effort (in terms of number of samples and the time between two subsequent samples) in order to obtain a reliable estimate of the buffer content distribution.

The next step is to insert the estimated variance into the dimensioning formulas of Section II. In Section VI, we do so for our reference set of real traces. We compare the resulting capacity by its "empirical counterpart," i.e., the minimum link rate such that, for the given trace, the performance requirement is met. We also systematically study the impact of the performance criterion on the required link rate and provide implementation guidelines. This section also indicates how our methodology carries over to a multilink setting.

Section VII concludes the paper. We discuss the results and the applicability of our approach. We also reflect on a number of related methods.

II. Generic Dimensioning Formulas

As explained in the introduction, an important prerequisite for dimensioning are formulas that determine the minimum required link rate for given characteristics of the offered traffic and performance target. 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 formulas that we derive under extremely weak conditions on the traffic process. The only substantial assumption is that we require that the traffic stream be stationary. With $A(s,t)$ denoting the amount of traffic arrived in $[s,t)$, it is assumed that 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 $A(t) := A(0,t)$. In this paper, we study dimensioning with respect to two performance criteria: "link transparency" and buffer overflow.

A. Link Transparency

In "link transparency," cf. [5], the main objective of bandwidth dimensioning 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: $\mathbb{P}(A(T) \geq \mathrm{C}T) \leq \varepsilon$. 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 service level agreements they agreed on with their clients.

Given the criterion $\mathbb{P}(A(T) \geq \mathrm{C}T) \leq \varepsilon$, we now derive a formula for the minimal link rate needed. Relying on the celebrated Chernoff bound, we have

$$\mathbb{P}(A(T) \geq \mathrm{C}T) \leq \inf_{\vartheta \geq 0} \left(e^{-\vartheta \mathrm{C} T} \mathbb{E} e^{\vartheta A(T)} \right), \text{ for any } \vartheta \geq 0.$$

As this inequality holds for *any* $\vartheta \geq 0$, we see that in order to be sure that $\mathbb{P}(A(T) \geq \mathrm{C}T) \leq \varepsilon$, it suffices to take the link rate C larger than

$$\mathrm{C}_1 := \inf_{\vartheta \geq 0} \frac{\log \mathbb{E} \exp(\vartheta A(T)) - \log \varepsilon}{\vartheta T}. \tag{1}$$

B. Buffer Overflow

The link-transparency criterion did not explicitly take into account the option of buffering packets. The distribution of the steady-state buffer content Q can be expressed in terms of the arrival process $A(\cdot)$: Reich's formula says that Q is distributed as the maximum of the process $A(-t,0) - \mathrm{C}t$:

$$Q =_{\mathrm{d}} \sup_{t \geq 0} (A(-t,0) - \mathrm{C}t)$$

where '$=_{\mathrm{d}}$' denotes equality in distribution. A second performance criterion could be to choose C such that $\mathbb{P}(Q > \mathrm{B}) < \varepsilon$, where B is the router's buffer size (alternatively, B/C can be interpreted as an upper bound on the delay); observe that Q depends on C.

In order to find the minimum required capacity, we need to characterize $\mathbb{P}(Q > \mathrm{B})$ as a function of C. This is done as follows. In the first place, observe that 'Reich' entails that $\mathbb{P}(Q > \mathrm{B})$ is the probability that $A(-t,0)$ exceeds $\mathrm{B}+\mathrm{C}t$ for some $t \geq 0$. This is a union of events (namely, the union over $t \geq 0$ of the

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

events $\{A(-t,0) \geq \mathrm{B} + \mathrm{C}t\}$); it often is accurate to approximate the probability of a union of events by the largest of the individual probabilities. We thus obtain

$$\begin{aligned}\mathbb{P}(Q > \mathrm{B}) &\equiv \mathbb{P}(\exists t : A(-t,0) \geq \mathrm{B} + \mathrm{C}t) \\ &\approx \sup_{t\geq 0} \mathbb{P}(A(-t,0) \geq \mathrm{B} + \mathrm{C}t).\end{aligned}$$

See, for instance, [14]; a further large-deviations justification for the "principle of the largest term" is given in [1] and [7]. It can be argued [15, Sec. 10.3] that the Chernoff bound is in fact a reasonable approximation, so we obtain

$$\mathbb{P}(Q > \mathrm{B}) \approx \sup_{t\geq 0} \inf_{\vartheta \geq 0} \left(e^{-\vartheta(\mathrm{B}+\mathrm{C}t)} \mathbb{E} e^{\vartheta A(-t,0)} \right). \tag{2}$$

Note that this approximation contains a conservative element (Chernoff bound) as well as an "aggressive" element (principle of the largest term), and it is not clear upfront which effect dominates. After some rearranging, we conclude that C should be at least (cf. [18] and [31, Eq. (5)])

$$\mathrm{C}_2 := \sup_{t\geq 0} \inf_{\vartheta \geq 0} \frac{\log \mathbb{E} \exp(\vartheta A(-t,0)) - \log \varepsilon - \vartheta \mathrm{B}}{\vartheta t}. \tag{3}$$

Clearly, application of (1) and (3) requires the estimation of the mgf $\mathbb{E}e^{\vartheta A(s,t)}$, with $\vartheta \geq 0$; in (1), just for $s = 0$ and $t = T$, but in (3), we even need $\mathbb{E}e^{\vartheta A(-t,0)}$ for all $t \geq 0$. Such an estimation is extremely demanding and far from straightforward (for specific situations, the results in [12] are helpful here; we comment on these in detail in Section VI). However, imposing some additional structure on $A(\cdot)$ may greatly simplify the dimensioning formulas; this structure should of course be flexible enough to still cover all relevant traffic patterns. The following example provides such a framework: the class of Gaussian processes.

Example: Suppose 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 $V(\cdot) : \mathbb{R}^+ \to \mathbb{R}^+$. The covariance structure is fully defined by the variance function, as it holds that $\mathbb{C}\mathrm{ov}(A(s), A(t)) = \frac{1}{2}(V(t) + V(s) - V(t-s))$. An important special case is *fractional Brownian motion*, in which $V(t)$ is proportional to t^{2H}, where $H \in (0,1)$ is the so-called Hurst parameter; when choosing $H = \frac{1}{2}$, we obtain 'ordinary' Brownian motion.

When assuming that traffic is Gaussian, with $\delta := \sqrt{-2\log\varepsilon}$, the dimensioning formulas (1) and (3) respectively reduce to

$$\mathrm{C}_1 = M + \frac{\delta}{T} \cdot \sqrt{V(T)} \tag{4}$$

$$\mathrm{C}_2 = M + \inf_{t\geq 0} \left(\frac{\delta}{t} \cdot \sqrt{V(t)} - \frac{\mathrm{B}}{t} \right). \tag{5}$$

Here, $A(-t,0)$ is distributed as $A(0,t)$ (as any Gaussian process is time-reversible), and $\mathbb{E}\exp(\vartheta A(t)) = M\vartheta t + \vartheta^2 V(t)/2$ (the latter identity follows, after some calculus, from $\int e^{-\frac{1}{2}x^2}\,\mathrm{d}x = \sqrt{2\pi}$, and completing the square). The important consequence of this is that for the application of the dimensioning formulas (1) and (3), it is not required anymore to estimate mgfs. Instead, they can be computed when we have estimates for the mean rate M and the variance curve $V(\cdot)$ at our disposal. In (1), we even need just $V(T)$ rather than the whole curve. □

III. Modeling Traffic Aggregates by Gaussian Processes

The example in the previous section showed that the dimensioning formulas become substantially more manageable for Gaussian inputs. In this section, we discuss traffic models in a general context, motivate why we concentrate on Gaussian models, and assess when Gaussian models are applicable.

A. Traffic Models

As argued earlier, a crucial prerequisite for using the dimensioning formulas (1) and (3) is the availability of simple yet accurate traffic models. Clearly, the simplest model with Poisson arrivals of packets has the undesirable feature that it fails to incorporate the (positive) correlations between packet arrivals as observed in real traces. For this reason, the model with (a superposition of) ON/OFF sources is an attractive alternative: a broad variety of correlation structures can be modeled by choosing appropriate distributions for the ON- and OFF-times. A variant of the latter model is the so-called $\mathrm{M}/\mathrm{G}/\infty$ input model, in which *flows* (groups of packets with some general distribution) arrive according to a Poisson process, remain in the system for some random time, and generate traffic during this sojourn time according to some (random or deterministic) pattern. By choosing a heavy-tailed flow-size distribution, strong positive correlations can be obtained. Two recent papers that fit these flow-level models are [2] and [3].

Rather than attempting to describe traffic at the flow level, one could also opt for trying to find models for the *aggregate traffic stream*. The development of this type of model was triggered by a number of measurement studies performed in the early 1990s, such as the famous Bellcore measurements [21], [30]. It was shown that, in many situations, the aggregate stream has self-similar properties and is long-range dependent (i.e., has a slowly decaying autocorrelation function).

A stochastic model, advocated by Norros in [28], [29], that has many desirable properties (e.g., long-range dependency) is fBm. fBm is a self-similar process (i.e., $A(ft)$ has the same distribution as $\phi(f)\,A(t)$, for $f \geq 0$, and a function $\phi(\cdot)$ that does not depend on t) and, as mentioned in the previous section, falls in the class of Gaussian models. In recent years, it found widespread use as a reference model for IP traffic; importantly, $H \in (\frac{1}{2}, 1)$ corresponds to long-range dependent, positively correlated traffic. By taking $V(t) = t^{2H}$, the dimensioning formulas (4) and (5) further simplify to

$$\begin{aligned}\mathrm{C}_1 &= M + \frac{\delta}{T^{1-H}} \\ \mathrm{C}_2 &= M + \delta^{1/H} \left(\frac{1-H}{\mathrm{B}} \right)^{1/H-1} H.\end{aligned}$$

Besides the above motivations for Gaussian traffic models, their applicability can also be explained from the Central Limit Theorem (CLT). The CLT entails that the sum of a large number of "small" independent (or weakly dependent), statistically more or less identical, random variables (users) has an approximately normal (i.e., Gaussian) distribution. Thus, one can expect that an aggregated traffic stream consisting of many individual communications may be modeled by a Gaussian stochastic process. However, it is clear that the CLT argumentation does not apply to any time-scale. On the time-scale of transmission of (minimum-size) packets, the traffic stream is always ON/OFF (either

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

there is transmission at link speed, or silence)—which is obviously not Gaussian. Thus, besides requiring that the number of users (referred to as "vertical aggregation") is sufficiently high, there should also be sufficient aggregation in time ('horizontal aggregation'). Kilpi and Norros [20] and Fraleigh *et al.* [14] pointed out the necessity of enough aggregation in both directions for traffic to be Gaussian; see also [13] for an early reference on horizontal and vertical aggregation.

We emphasize that $\mathrm{M}/\mathrm{G}/\infty$ input models and Gaussian models do not exclude one another. As long as the aggregation level is sufficiently high, both models could fit very well. The intuitive reason for this is that, under those circumstances, a Poisson random variable can be approximated accurately by a normal random variable. In light of this, it is not surprising one can formally prove that in a particular limiting regime (more precisely, by speeding up the arrivals), the $\mathrm{M}/\mathrm{G}/\infty$ input model converges to a Gaussian process; this can be proven as in [10].

We remark, however, that for the $\mathrm{M}/\mathrm{G}/\infty$ input model, it may take a substantial effort to fit all the parameters corresponding to the flow duration, traffic transmission rate, etc.; see [3].

B. Gaussianity for Different Levels of Horizontal and Vertical Aggregation

We now report on our findings regarding Gaussianity. These are largely in line with the conclusions in [14] and [20]; more experiments can be found in [25] and [26]. We have tried to make the data sets as representative as possible. The locations cover all sorts of users. As we largely follow the methodology of [14] and [20], we have attempted to keep this subsection brief.

The goal of this subsection is to provide empirical support for the claim that the Gaussianity assumption is justified at many locations, with very distinct types of user. The five locations we have considered are: (U) a university residential network (15 traces, 1800 hosts); (R) a research institute (185 traces, 250 hosts); (C) a college network (302 traces, 1500 hosts); (A) an ADSL access network (50 traces, 2000 hosts); and (S) a server hosting provider (201 traces, 100 hosts). Each trace relates to 15 min (real time). At these locations, traffic is generated at average (aggregate) rates of 170, 6, 35, 120, and 12 Mb/s, respectively.

Both [14] and [20] observe that there should be a sufficient level of aggregation to make traffic Gaussian; [14] further quantifies this claim by reporting that at least a traffic rate of 50 Mb/s is needed (when considering backbone links). We now verify whether this applies for our data set {U, R, C, A, S}. QQ-plots display the (empirical) quantiles of the distribution under consideration against the quantiles of some test distribution (i.e., a Gaussian distribution, in our case). A good fit to the Gaussian distribution means that all points are close to a straight line. Therefore, as a measure of Gaussianity, the *linear correlation coefficient* can be used as the goodness-of-fit measure, as in [17] and [20]; in [24], it is shown that the use of this statistic leads to similar conclusions as the Kolmogorov–Smirnov test that was used in [14]. Fig. 1, which focuses on the time-scale of 1 s, confirms the findings of [14]: roughly spoken, for this time-scale, an average traffic rate of some tens of Mb/s seems enough to safely assume Gaussianity.

Suppose we observe that a specific traffic stream is fairly Gaussian at a certain time-scale. One may then wonder what this says about Gaussianity at other time-scales. If Gaussianity would be (more or less) preserved across time-scales, then one needs to verify Gaussianity on just one time-scale. It is clear that this reasoning has its limitations; recall that traffic is certainly *not* Gaussian at very small time-scales, and also Gaussianity will be lost for very large time-scales.

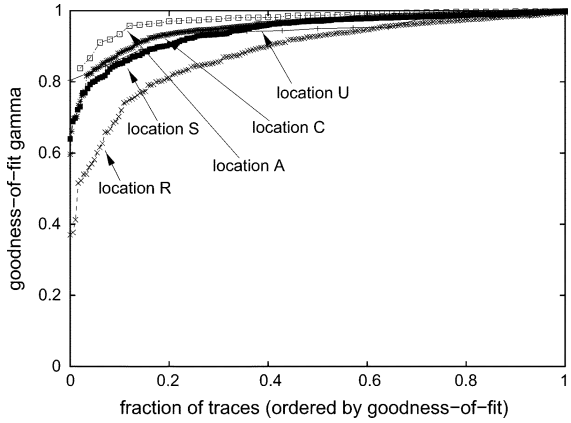

Fig. 1. Distribution of the linear correlation coefficient over all measurements.

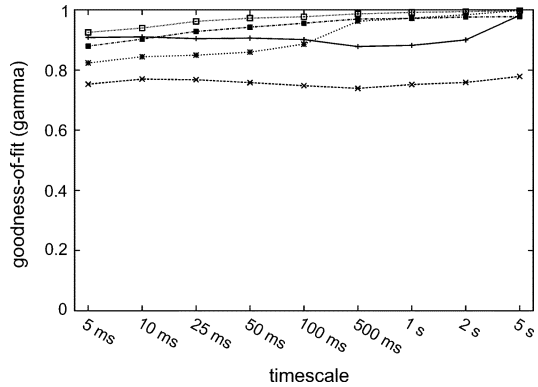

Fig. 2. Linear correlation coefficient over five measurements from location R.

First, we look at an example with only a few traces. We determine the linear relation coefficient γ at nine time-scales τ_i, ranging from 5 ms to 5 s. The results, based on five traces from measurement location R, are plotted in Fig. 2. As reflected by the more or less horizontal lines, the graph suggests that the Gaussianity is rather constant across time-scales.

Next, we investigate this for all traces. We introduce ν_γ as measure of the 'variation of the linear correlation coefficient,' defined as the square root of the sample variance of the γ_{τ_i}

$$\nu_\gamma := \sqrt{\widehat{\mathbb{V}\mathrm{ar}}\left(\gamma_{\tau_1}, \gamma_{\tau_2}, \ldots, \gamma_{\tau_9}\right)}.$$

The interpretation is that when ν_γ is low, the traffic is (more or less) equally Gaussian (or non-Gaussian) across multiple time-scales.

We have computed ν_γ using all traces from each measurement location. After ordering them from low to high values of ν_γ, they are plotted in Fig. 3. Clearly, ν_γ is small in most cases; in over 95% of the traces, ν_γ is below 0.05. Thus, we may conclude that γ is quite constant over different time-scales. In other words, traffic that exhibits Gaussian characteristics at one time-scale, is likely to be Gaussian at other time-scales as well (for the time-scales that we investigated, and bearing in mind the limitations mentioned above). In addition, in line with the conclusions of [14], we see from our experiments that

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

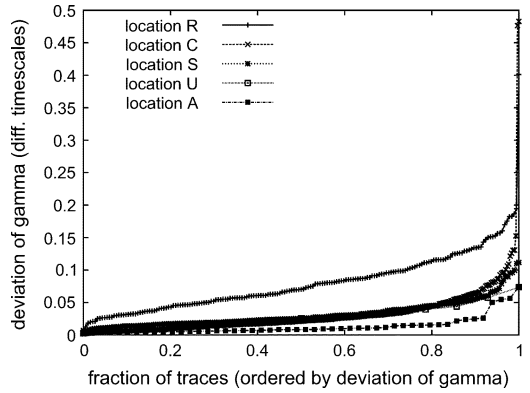

Fig. 3. Variance of linear correlation coefficient over time, over all measurements at various time-scales, at locations {U, R, C, A, S}.

Gaussianity usually applies from the ms-level on (In [14], even smaller timescales were considered, but these appear to be irrelevant for dimensioning purposes.).

It is important to notice that in some cases, our experiments show that the Gaussian model is *not* accurate—for instance, a substantial part of the traces from location R, where there is a relatively low aggregation level. As our dimensioning approach relies on the Gaussianity assumption, it is important to verify whether it still gives reasonable outcomes for situations in which the traffic is *not* Gaussian. We come back to this issue in detail in Section V.

IV. Estimation of the Mean Traffic Rate and Variance Curve

In Section II we derived the dimensioning formulas (1) and (3), which require knowledge of the mean traffic rate M and the variance curve $V(\cdot)$. As argued earlier, M can be determined by standard coarse-grained traffic measurements (e.g., polling Interfaces Group MIB counters via SNMP every 5 min). It is clear that determining the variance curve $V(\cdot)$ is more involved. The standard way to estimate $V(T)$ (for some given interval length T) is what we refer to as the "direct approach": perform traffic measurements for disjoint intervals of length T and just compute their sample variance. It is noted that the convergence of this estimator could be prohibitively slow when traffic is long-range dependent [4, Ch. I], but the approach has two other significant drawbacks.

- When measuring traffic using time windows of duration T, it is clearly possible to estimate $V(T)$, $V(2T)$, $V(3T)$, etc. However, these measurements obviously do not give any information on $V(\cdot)$ on time-scales *smaller* than T. Hence, to estimate $V(T)$, measurements should be done at granularity T or less. This evidently leads to a substantial measurement effort.
- The dimensioning formula (3) requires knowledge of the *entire* variance function $V(\cdot)$, whereas the direct approach described above just yields an estimate of $V(T)$ on a prespecified timescale T. Therefore, a method that estimates the entire curve $V(\cdot)$ is preferred.

This section presents a powerful alternative to the direct approach; we refer to it as the *inversion approach*, as it "inverts" the buffer content distribution to the variance curve. This inversion approach overcomes the problems identified above. We rely on the large-deviations framework of Section II for Gaussian inputs, as justified in Section III.

A. Inversion Formula

In this subsection, we first show how, for given variance curve $V(\cdot)$ (and mean rate M), the probability $\mathbb{P}(Q > \mathrm{B})$ can be approximated. Then, this explicit formula is used to "invert" this relation. We establish a formula for $V(t)$, given the (complementary) buffer content distribution $\mathbb{P}(Q > \cdot)$.

- In Section II, we found the approximation (2). For the special case of Gaussian traffic, using $\mathbb{E}\exp(\vartheta A(t)) = M\vartheta t + \vartheta^2 V(t)/2$, the minimization over ϑ can be explicitly evaluated. We find that (2) reduces to

$$\mathbb{P}(Q > \mathrm{B}) \approx e^{-I(\mathrm{B})}, \text{ with } I(\mathrm{B}) := \inf_{t>0} \frac{((\mathrm{C}-M)t+\mathrm{B})^2}{2V(t)}. \tag{6}$$

- Supposing that approximation (6) is exact, it implies that

$$\forall \mathrm{B} \geq 0 : \forall t \geq 0 : -\log \mathbb{P}(Q > \mathrm{B}) \leq \frac{((\mathrm{C}-M)t+\mathrm{B})^2}{2V(t)}$$

or, equivalently

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

Clearly, the latter inequality implies that for all $t \geq 0$

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

In fact, this upper bound on the variance is under mild conditions tight in a large-deviations sense (i.e., in the many-sources framework); for details, we refer to [22, Theorem 4]. Remarkably, this says that, loosely speaking, for Gaussian traffic the *buffer content distribution uniquely determines the variance function*, and it does so through the explicit formula

$$V(t) \approx \inf_{\mathrm{B}\geq 0} \frac{(\mathrm{B}+(\mathrm{C}-M)t)^2}{-2\log \mathbb{P}(Q > \mathrm{B})}. \tag{7}$$

Hence, if we can estimate $\mathbb{P}(Q > \mathrm{B})$, then the "inversion formula" (7) can be used to retrieve the variance. Notice that the infimum can be computed for any t, and consequently we get an approximation for the entire variance curve $V(\cdot)$ (of course up to some finite horizon).

Remark: Observe that in fact $\mathbb{P}(Q > \mathrm{B})$ is also a function of the service speed C (as the random variable Q depends on C). Interestingly, as the variance of the offered traffic in a window of length t, $V(t)$, does not depend on C, (7) entails that, if the approximation is correct, the minimum in the right-hand side should also not depend on C.

The unique correspondence between input and buffer content does not only apply for Gaussian processes, but can be found in many other situations as well. A well-known example is the M/G/1 queue. Assume Poisson arrivals stream (rate λ) of jobs, with service requirement distributed as random variable B (where its Laplace transform is denoted by $b(s) := \mathbb{E}e^{-sB}$) and service speed C. Let the load ϱ be defined as $\lambda\,\mathbb{E}B$.. Denote by $q(\cdot;\mathrm{C})$ the Laplace transform of the buffer content: $q(s;\mathrm{C}) :=$

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

$\mathbb{E}e^{-sQ}$. The subscript "C" is added to emphasize the dependence on C. The Pollaczek–Khintchine formula says

$$q(s;\mathrm{C}) = \left(1 - \frac{\varrho}{\mathrm{C}}\right) \cdot \frac{s}{s - (\lambda/\mathrm{C}) \cdot (1 - b(s))}.$$

This can be inverted to

$$b(s) = \frac{\mathrm{C}s}{\lambda q(s;\mathrm{C})} \cdot \left(\left(1 - \frac{\varrho}{\mathrm{C}}\right) + \left(\frac{\lambda}{\mathrm{C}s} - 1\right) q(s;\mathrm{C})\right).$$

We see similar phenomena as above: 1)The buffer-content distribution uniquely defines the distribution of the input process (the distribution of B). 2) The left-hand side of the previous display clearly does not depend on C, so apparently the C in the right-hand side also cancels. □

Example: Brownian Bridge: In a Brownian bridge, time is restricted to the interval $[0, 1]$, and $V(t) = t(1 - t)$. It is easily verified that

$$\arg \inf_{t\in(0,1)} \frac{((\mathrm{C} - M)t + \mathrm{B})^2}{2t(1-t)} = \frac{\mathrm{B}}{(\mathrm{C} - M) + 2\mathrm{B}}$$
$$I(\mathrm{B}) = 2\mathrm{B}(\mathrm{B} + (\mathrm{C} - M)).$$

We can now derive the upper bound on $V(t)$ from $I(\mathrm{B})$. To this end, we first compute

$$\arg \inf_{\mathrm{B}\geq 0} \frac{((\mathrm{C} - M)t + \mathrm{B})^2}{2I(\mathrm{B})} = \frac{(\mathrm{CC} - M)t}{1 - 2t}$$

then a lengthy calculation indeed gives $V(t) \leq t(1 - t)$, as desired. □

B. Algorithm for Estimating Variance Through Inversion

In this section, we show how the inversion formula (7) can be used to estimate $V(\cdot)$. This inversion procedure consists of two steps. First, we estimate the (complementary) buffer content distribution $\mathbb{P}(Q > \cdot)$, which is in the sequel abbreviated to BCD. Then, we "invert" the BCD to the variance curve $V(\cdot)$ by applying (7). We propose the following algorithm:

Algorithm: Inversion

1. Collect 'snapshots' of the buffer contents: $q_1, \ldots, q_N$; here, q_i denotes the buffer content as measured at time $\tau_0 + i\tau$, for some $\tau > 0$. Estimate the BCD by the empirical distribution function of the q_i, i.e., estimate $\mathbb{P}(Q > \mathrm{B})$ by $\phi(\mathrm{B}) := \#\{i : q_i > \mathrm{B}\}/N$.
2. Estimate $V(t)$ by

$$\inf_{\mathrm{B}\geq 0} \frac{(\mathrm{B} + (\mathrm{C} - M)t)^2}{-2\log\phi(\mathrm{B})}$$

for any $t \geq 0$. □

Clearly, to obtain an accurate estimate of the BCD, both τ and N should be chosen sufficiently large. We come back to this issue in Section V.

C. Demonstration of Inversion Procedure

In the remainder of this section, we demonstrate the inversion approach through a simulation with synthetic (fBm) input; we emphasize, however, that the procedure could be performed for any other traffic process. We compare the estimated variance curve with the (known) actual variance curve. In this way, we get a first impression of the accuracy of our approach; a more detailed numerical evaluation follows in Section V.

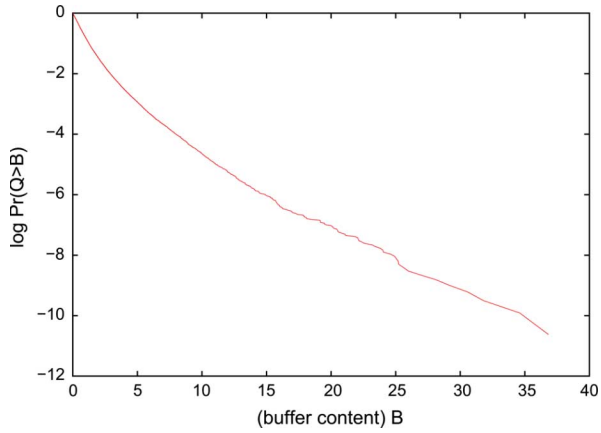

Fig. 4. Sample BCD.

Consider a queue in discrete time with link rate C fed by fBm. Its buffer dynamics are simulated as follows. A simulator [11] is used to generate fBm with a specific Hurst parameter $H \in (0, 1)$, yielding a list of numbers that represent the "offered traffic per time slot." These numbers serve as input to the queue. For every slot, the amount of offered traffic is added to the buffer content, while an amount equal to C is subtracted from the buffer content (where, when this number becomes negative, it is put to zero). Then, every $\tau \in \mathbb{N}$ slots, the queue's content is observed, yielding N snapshots $q_1, \ldots, q_N$ that are used to estimate $\mathbb{P}(Q > \cdot)$ (as in the above algorithm).

In this demonstration of the inversion procedure, we generate a fBm traffic trace with Hurst parameter $H = 0.7$; we take standard fBm, i.e., $M = 0$ and $V = t^{2H}$. The link capacity C is set to 0.8. The trace consists of 2^{24} slots, and we take snapshots of the buffer content every $\tau = 128$ slots.

We now discuss the output of the inversion procedure for our simulated example with fBm traffic. First, we estimate the BCD; a plot is given in Fig. 4. For presentation purposes, we plot the logarithm of the BCD, i.e., $\log \mathbb{P}(Q > \mathrm{B})$. The BCD in Fig. 4 is "less smooth" for larger values of B, which is due to the fact that large buffer levels are rarely exceeded, leading to less accurate estimates.

Second, we estimate the variance $V(t)$ for t equal to the powers of 2 ranging from 2^0 to 2^7 using the BCD, i.e., by using the algorithm. The resulting variance curve is shown in Fig. 5 ("inversion approach"). The minimization (over B) was done by straightforward numerical techniques. To get an impression of the accuracy of the inversion approach, we have also plotted the variance curve as can be estimated directly from the synthetic traffic trace (i.e., by using the 'direct approach' introduced earlier), as well as the real variance function for fBm traffic, i.e., $V(t) = t^{2H}$. The figure shows that the three variance curves are remarkably close to each other. This confirms that the inversion approach is an accurate way to estimate the burstiness. We note that the graph shows that the inversion approach slightly *overestimates* the variance. A more detailed validation of the inversion approach follows in the next section.

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

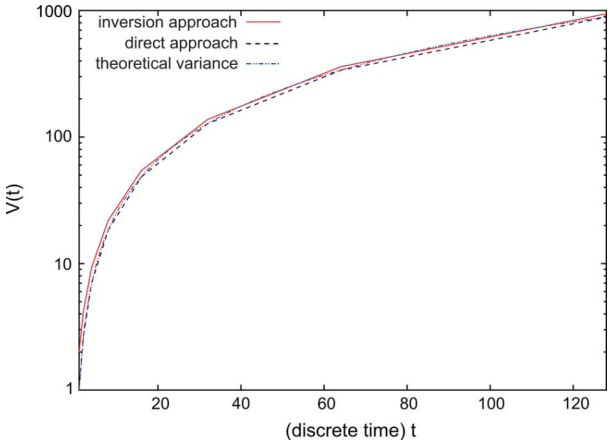

Fig. 5. Sample variance curves.

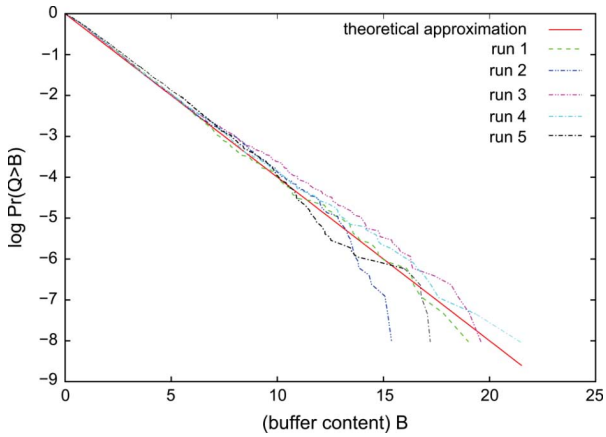

Fig. 6. $\mathbb{P}(Q > \mathrm{B})$ and theoretical approximation ($H = 0.5$).

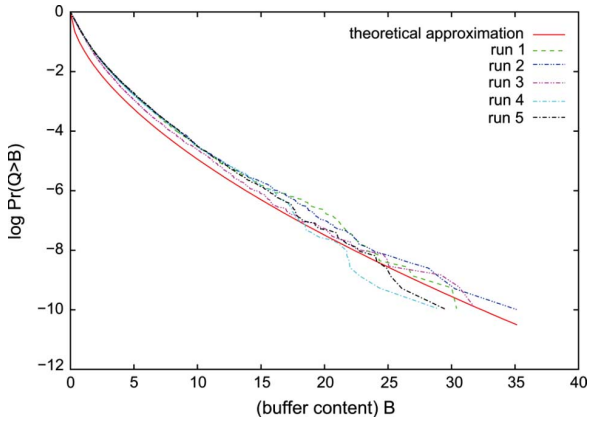

Fig. 7. $\mathbb{P}(Q > \mathrm{B})$ and theoretical approximation ($H = 0.7$).

V. Error Analysis of the Inversion Procedure

In the previous section, the inversion approach was demonstrated, and it was shown to perform well for fBm with $H = 0.7$ under a specific choice of N and τ. Evidently, the key question is under what circumstances the procedure works. To this end, we first identify the three possible sources of errors:

- The inversion approach is based on the *approximation* (7).
- The $\mathrm{BCD} > \mathbb{P}(Q > \cdot)$ is *estimated*; there could still be an estimation error involved. In particular, one may wonder what the impact of the choice of N and τ is.
- The procedure *assumes* "perfectly Gaussian" traffic, although real network traffic may not be (accurately described by) Gaussian; see Section III.

We will now quantitatively investigate the impact of each of these errors on our 'inversion approach.'

A. Approximation of the Buffer Content Distribution

In (6), an approximation of the BCD is given. As the inversion formula (7) is based on this approximation, evidently errors in (6) might induce errors in the estimate of $V(\cdot)$. Therefore, we now assess the accuracy of (6). We focus on the practically relevant case of fBm; in line with the previous section, we choose $M = 0$ and $V(t) = t^{2H}$. Straightforward calculations now reveal that (6) implies

$$\log \mathbb{P}(Q > \mathrm{B}) \approx -\frac{1}{2} \cdot \left(\frac{\mathrm{B}}{(1-H)} \right)^{2-2H} \cdot \left(\frac{\mathrm{C}}{H} \right)^{2H}.$$

We verify how accurate the approximation is, for two values of H: the Brownian case $H = 0.5$ and a case with long-range dependence $H = 0.7$ (which is a rather typical value, as found in many measurement studies). Several runs of fBm traffic are generated (with different random seeds), with 2^{24} slots of traffic per run, which are used to simulate the buffer dynamics. For $H = 0.5$, we choose link rate $\mathrm{C} = 0.2$; for $H = 0.7$ we choose $\mathrm{C} = 0.8$. These choices of C are such that the queue is nonempty sufficiently often (to make sure that a reliable estimate of the BCD is obtained). Figs. 6 and 7 show for the various runs the approximation of the BCD, as well as their theoretical counterpart. Particularly for small B, the empirically determined BCD matches very well with the values predicted by (6).

B. Estimation of the Buffer Content Distribution

The inversion formula requires the BCD $\mathbb{P}(Q > \cdot)$, which we approximate by $\phi(\cdot)$, i.e., its empirical counterpart. This may lead to errors; the impact of this error on the estimation of the variance curve is the subject of this subsection. It could be expected that the larger N (more observations) and τ (less correlation between the observations), the better the estimate.

We first investigate the impact of N. The simulator is run as in previous cases (with $H = 0.7$) with the difference that we only use the first $x\%$ of the snapshots samples to estimate the BCD. Fig. 8 shows the estimation of the buffer content distribution for various x ranging from 0.1% to 100%. The figure shows that, particularly for relatively small B, a relatively small number of observations suffices to obtain an accurate estimate.

Notice that we chose in our inversion procedure a *fixed* sampling frequency (i.e., τ^{-1}). It can be seen that this "periodic sampling" is by no means necessary; the BCD-estimation procedure obviously still works when the sampling epochs are not equally spaced. In fact, one should realize that, if the sampling is performed in a purely periodic fashion, and if in addition (a substantial part of) the traffic is also periodic, then one may obtain even unreliable estimates. Therefore, it may have advantages to sample at, for instance, Poisson epochs.

Second, we investigate the impact of the interval length between two consecutive snapshots τ. Fig. 9 shows the determined

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

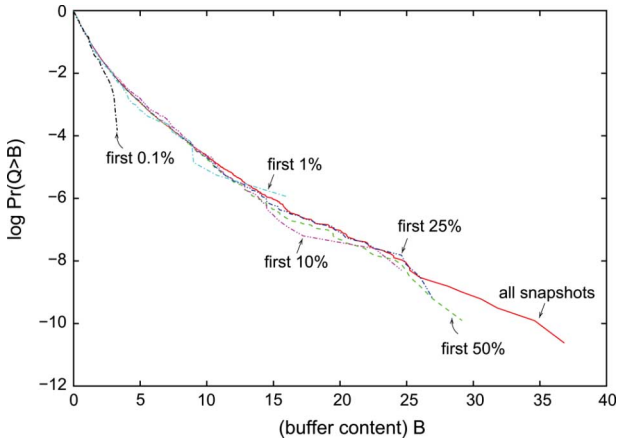

Fig. 8. Comparing $\mathbb{P}(Q > \text{B})$ for various trace lengths.

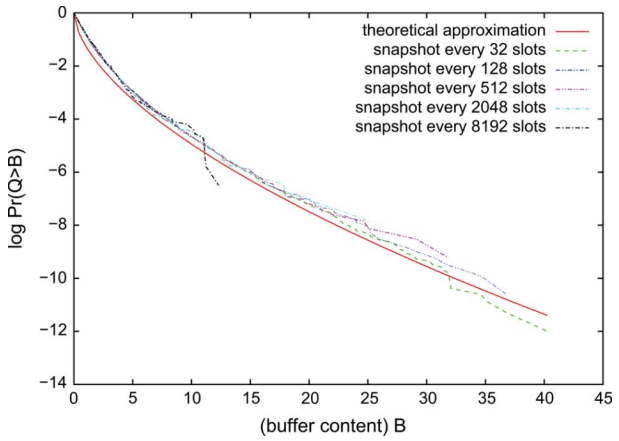

Fig. 9. Comparing $\mathbb{P}(Q > \text{B})$ for various sampling intervals.

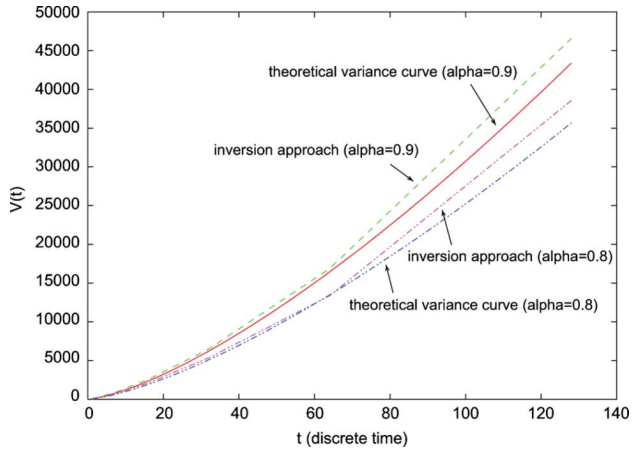

Fig. 10. Variance curves for Gaussian/non-Gaussian traffic mixtures,$\alpha = 0.9$ and $\alpha = 0.8$.

BCD for τ ranging from observing every 32 to every 8192 slots. It can be seen that, particularly for small B, the fit is quite good, even when the buffer content is polled only relatively rarely.

C. The Impact of the Gaussianity Assumption

Approximation (6) explicitly assumes that the traffic process involved is Gaussian. We have seen in Section III that this claim is not always justified. Therefore, we now investigate how sensitive our inversion approach is with respect to the Gaussianity of the input traffic.

We study the impact of non-Gaussianity by mixing, for every slot, a fraction α of the generated fBm traffic with a fraction $1-\alpha$ traffic from an alternative (non-Gaussian) stream before the mixture is fed into the queue. Note that the variance of the mixture is, in self-evident notation

$$V(t) = \alpha^2 V_{[\text{fBm}]}(t) + (1-\alpha)^2 V_{[\text{alt}]}(t).$$

We vary α from 1 to 0 to assess the impact of the non-Gaussianity.

The alternative input model that we choose here is the $\text{M}/\text{G}/\infty$ input model (see Section III), inspired by [1]–[3]. We denote the flow arrival rate by (λ). While in the system, traffic is generated at a constant rate r. In line with measurements studies (a classical reference is [9]), we choose Pareto (β) jobs, obeying the distribution function

$$F(x) = 1 - 1/(x+1)^{\beta}, \quad \beta > 0.$$

As the objective is to assess the impact of varying the parameter α, we have chosen to select the parameters of the $\text{M}/\text{G}/\infty$ input model such that it is "compatible" with fBm in that their means are equal and their variances $V(\cdot)$ are "similar" (in a sense that is defined below). This is achieved as follows.

- The means of both traffic stream are made compatible by adding a drift to the fBm inputs equal to the mean of the $\text{M}/\text{G}/\infty$ traffic stream, i.e., $\lambda r/(\beta-1)$. The Gaussianity of the fBm input is not affected by the addition of such a drift.
- To make the variances of both traffic streams "compatible," we make use of a derivation in earlier work of the exact variance function $V(t)_{[\text{alt}]}$. See [23]; there it was shown that for the relevant case [9] of $\beta \in (1,2)$, for large t

$$V_{[\text{alt}]}(t) \approx 2r^2 \frac{\lambda t^{3-\beta}}{(3-\beta)(2-\beta)(\beta-1)}.$$

  It is noted that long-range dependence is a property of long time-scales. We therefore chose to select the parameters of the $\text{M}/\text{G}/\infty$ input model such that the ratio of the variances converges to a 1 as t goes ∞, thus guaranteeing that they possess the same "degree of long-range dependence."

Clearly, we can now estimate the remaining parameters and compute the variance of the traffic mixture.

The next step is to run the simulation for different values of α. We then determine the (theoretical) variance curve of the traffic mixture and compare it to the variance curve found through the inversion approach. In Fig. 10, we focus on the "nearly Gaussian" cases $\alpha = 0.8$ and $\alpha = 0.9$, which are plotted together with their theoretical counterparts. The figure shows that the presence of non-Gaussian traffic has some, but no crucial, impact on our inversion procedure. Note that the non-Gaussian traffic may "have some Gaussian characteristics," as argued in Section III.

In the (extreme) case of $\alpha = 0$, i.e., no Gaussian traffic at all, Fig. 11 shows that the fit is substantially degraded. The graph shows the theoretical variance curve, the curve based on the "direct approach," as well as the curve based on the inversion approach.

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

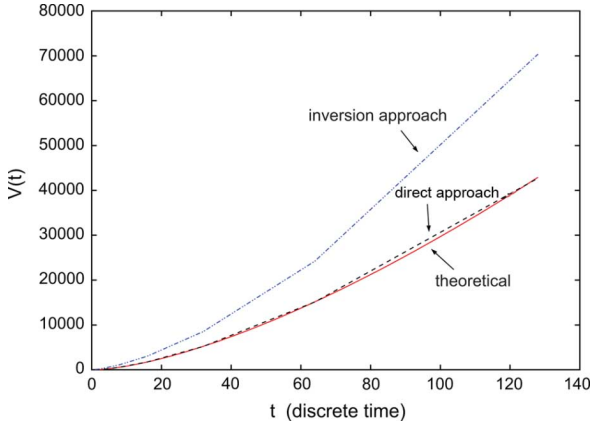

Fig. 11. Variance curves for Gaussian/non-Gaussian traffic mixture, $\alpha = 0$.

We conclude that our simulation experiments show the "robustness" of the inversion procedure. Despite the approximations involved, with a relatively low measurement effort, the variance curve is estimated accurately—even for traffic that is not "perfectly Gaussian." Given the evident advantages of the inversion approach over the 'direct approach' (minimal measurement effort required, retrieval of the entire variance curve $V(\cdot)$, etc.; see the discussion in Section III), the former method is to be preferred. In the next section, we verify whether this conclusion also holds for real (i.e., not artificially generated) network traffic. This is done by inserting the estimated mean and variance into the dimensioning formulas so that we can validate whether the resulting bandwidth values are such that the performance requirement is met.

VI. Bandwidth Dimensioning Procedure and Validation

As we have seen in Section V, the inversion approach shows rather good performance: Even if the underlying traffic deviates from Gaussian, one still obtains a relatively good estimate of the variance function. It remains unclear, however, whether plugging in this variance function into the dimensioning formula, i.e., (4) or (5), also leads to good estimates of the required bandwidth. Another question is whether such conclusions remain valid for the traces from our data set (rather than artificially generated data). The purpose of this section is to study these issues. We begin by detailing our dimensioning approach.

A. Dimensioning Procedure

We now describe our procedure to estimate the bandwidth needed in order to meet a predefined performance criterion. Suppose we have a trace of data, consisting of timestamps of packets, as well as the corresponding packet sizes; recall that each trace in our data set corresponds to 15 min (real time). We wonder what the minimum service rate is such that the performance requirement is satisfied.

Algorithm: Dimensioning

1. Estimate the variance function $V(\cdot)$ by estimating the BCD and performing algorithm INVERSION.
2. Insert this estimated variance into the dimensioning formula, i.e., (4) or (5). □

Notice that in the above procedure there was one choice left open. In the first step, the BCD is determined by feeding the trace into the queue, and we did not specify the service speed, say c_q, of this queue. Clearly, c_q should not be chosen too high because then the queue would be nearly always empty, leading to poor estimates of the BCD; we return to this issue later.

B. Validation

The idea behind this validation section is to systematically assess the accuracy of our dimensioning approach. We do so by decoupling two effects.

— *Validation of the required bandwidth formula*. Suppose we are given perfect information about the variance function, and we plug this into our dimensioning formula. Here, the question is: How good is the resulting estimate for the required bandwidth?

— *Impact of estimation errors in $V(\cdot)$ on required bandwidth*. Here, we test how the errors in the estimate of $V(\cdot)$, caused by the inversion procedure, have impact on the estimate of the required bandwidth.

The "decoupling" allows us to gain precise insight into—and thus a proper validation of—both steps of the above link-dimensioning procedure. We focus on the criterion of link transparency and the corresponding dimensioning formula (4); with the same methodology, one can perform the validation with respect to the buffer overflow criterion.

1) Validation of the Required Bandwidth Formula: We now check the accuracy of the dimensioning formula (4). It requires knowledge of M and $V(T)$. As we do not have their "real values," we estimate them using the 'direct approach.' We emphasize that this direct approach has significant disadvantages in practice (see the discussion in Section IV), but in order to assess the accuracy of (4), we do not have any alternative.

In more detail, from each trace we estimate the average traffic rate M and the variance $V(T)$ of the offered traffic at time-scale T. With A_i denoting the amount of traffic offered over the ith interval of length T

$$\widehat{M} = \frac{1}{nT}\sum_{i=1}^{n} A_i \quad \text{and} \quad \widehat{V}(T) = \frac{1}{n-1}\sum_{i=1}^{n}(A_i - M)^2.$$

Then, the resulting estimates, as well as the specified values of T and ε, are inserted into (4) to obtain the (estimated) required bandwidth.

We choose to determine the average traffic rate M per 15 min (recall that each trace contains 15 min of traffic) and set T to 1 s, 500 ms, and 100 ms (and thus determine the variance at those timescales), which are, for various applications, time-scales that are important to the perception of quality by (human) users. We set ε to 1%. Importantly, note that these settings for T and ε are just examples; network providers can choose the setting that suits their (business) needs best.

In order to validate if the estimated bandwidth capacity c indeed 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 intervals of length T, in which the amount of offered traffic A_i exceeds the estimated required capacity cT—we stress the fact that 'exceedance' in this context does not correspond to "packet loss." In other words

$$\hat{\varepsilon} := \frac{\#\{A_i \mid A_i > cT\}}{n} \quad (i \in 1 \ldots n).$$

If c is properly dimensioned, then "exceedance" (as in $A(T) > cT$) may be expected in a fraction ε of all intervals.

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

TABLE I
REQUIRED BANDWIDTH: ESTIMATION ERRORS AND DIMENSIONING FACTOR ($\varepsilon = 0.01$)

	T	avg $\hat{\varepsilon}$	avg $\lvert\varepsilon - \hat{\varepsilon}\rvert$	stderr $\lvert\varepsilon - \hat{\varepsilon}\rvert$	d	σ_d
U	1 s	0.0115	0.0095	0.0067	1.33	0.10
	500 ms	0.0115	0.0089	0.0067	1.35	0.09
	100 ms	0.0104	0.0077	0.0047	1.42	0.09
R	1 s	0.0137	0.0062	0.0060	2.91	1.51
	500 ms	0.0145	0.0063	0.0064	3.12	1.57
	100 ms	0.0132	0.0050	0.0053	3.82	1.84
C	1 s	0.0072	0.0069	0.0047	1.71	0.44
	500 ms	0.0072	0.0066	0.0043	1.83	0.49
	100 ms	0.0075	0.0055	0.0041	2.13	0.67
A	1 s	0.0019	0.0083	0.0027	1.13	0.03
	500 ms	0.0019	0.0083	0.0024	1.14	0.03
	100 ms	0.0022	0.0079	0.0020	1.19	0.03
S	1 s	0.0102	0.0052	0.0050	1.98	0.78
	500 ms	0.0102	0.0049	0.0055	2.10	0.87
	100 ms	0.0112	0.0040	0.0059	2.44	1.01

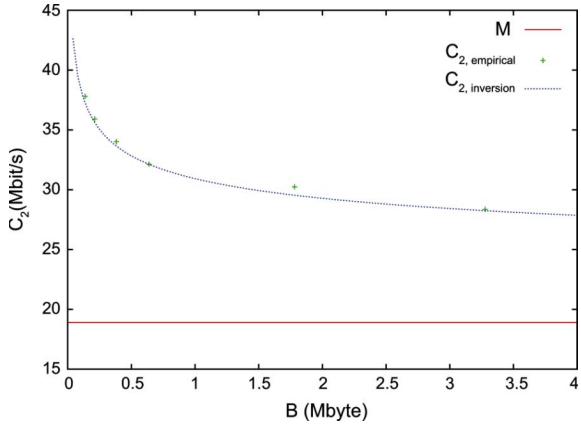

Fig. 12. Required bandwidth as a function of the buffer size B for a trace at location R. The '+' are the empirical values, and the line is the curve using the estimate of $V(\cdot)$ obtained by inversion; $\varepsilon = 0.01$.

There are, however, (at least) two reasons why $\hat{\varepsilon}$ and ε may not be equal in practice: 1) First, (4) assumes "perfectly Gaussian" traffic, which is, as we have seen, not always the case. Evidently, deviations from 'perfectly Gaussian' traffic may have an impact on the estimated C. 2) Second, 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. 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 I presents the average differences between the targeted ε and the "realized exceedance" $\hat{\varepsilon}$ at each location, as well as the standard deviations, for three different time-scales T. The table shows that *per trace* the differences are modest: at most in the order of the target probability $\varepsilon = 0.01$ (see column avg $\lvert\varepsilon - \hat{\varepsilon}\rvert$). Dimensioning decisions, however, are likely to be based on several traces at the same location (rather than just one trace), and we therefore also included the column with the per-location average of the $\hat{\varepsilon}$. These turn out to be close to the target probability and, in some cases, even substantially less.

When dimensioning a network link, network providers often use "rules of thumb," such as $\mathrm{C} = \alpha M$, for a given number α (for instance, take the mean rate, increased by 50%). To verify whether such a (simplistic) approach could work, it is interesting to get an idea of the required "dimensioning factor," that is, the (estimated) required bandwidth capacity compared to the average load (i.e., C/M). These dimensioning factors and their standard deviations, averaged over all traces at each location, are given in the two rightmost columns of Table I, for $T = 1$ s, 500 ms, and 100 ms. It 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 time-scale of $T = 1$ s. 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 Mb/s with a 1 Gb/s backbone). At location U, the user aggregation level is much higher, and hence, the traffic aggregate is "more smooth." Our conclusion is that simplistic dimensioning rules of the type $\mathrm{C} = \alpha M$ are inaccurate, as the α 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 dimensioning factors (cf. Table I) for the present case studies can be obtained as the ratio of C and M at certain T and ε. As indicated, the dimensioning factor increases when the performance criterion (through ε and T) becomes more stringent. To give a few examples of the impact of the performance parameters T and ε on the required bandwidth capacity, we plot curves for the required bandwidth capacity at $T = 10, 50, 100$ and 500 ms and ε ranging from 0.00001 to 0.1 in Fig. 13. In these curves, M and $V(T)$ are (directly) estimated from an example traffic trace collected at each of the locations {U, R, C, A, S}.

Fig. 13 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 Fig. 13, i.e., location U, example trace #1. At $\varepsilon = 10^{-5}$, the difference in required bandwidth between $T = 10$ ms and $T = 100$ ms is some 20%. At $T = 100$ ms, the difference in required bandwidth between $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-4}$ is just 3% approximately.

We have verified whether the required bandwidth 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 I). 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 $T = 500$ ms, for instance, there are only 1800 samples in our traffic trace (which has a length of 15 min), 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 of concurrent users) combined with a small time-scale of $T = 10$ ms, the traffic is no longer Gaussian (i.e., $\gamma \ll 1$). Consequently, the accuracy of our required bandwidth estimation decreases.

2) Impact of Estimation Errors in the Variance on the Required Bandwidth: In Section V, we have seen for artificially generated traffic that the inversion worked well, and the first part of the present section assessed the quality of the bandwidth dimensioning formula (4). We now combine these two elements:

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

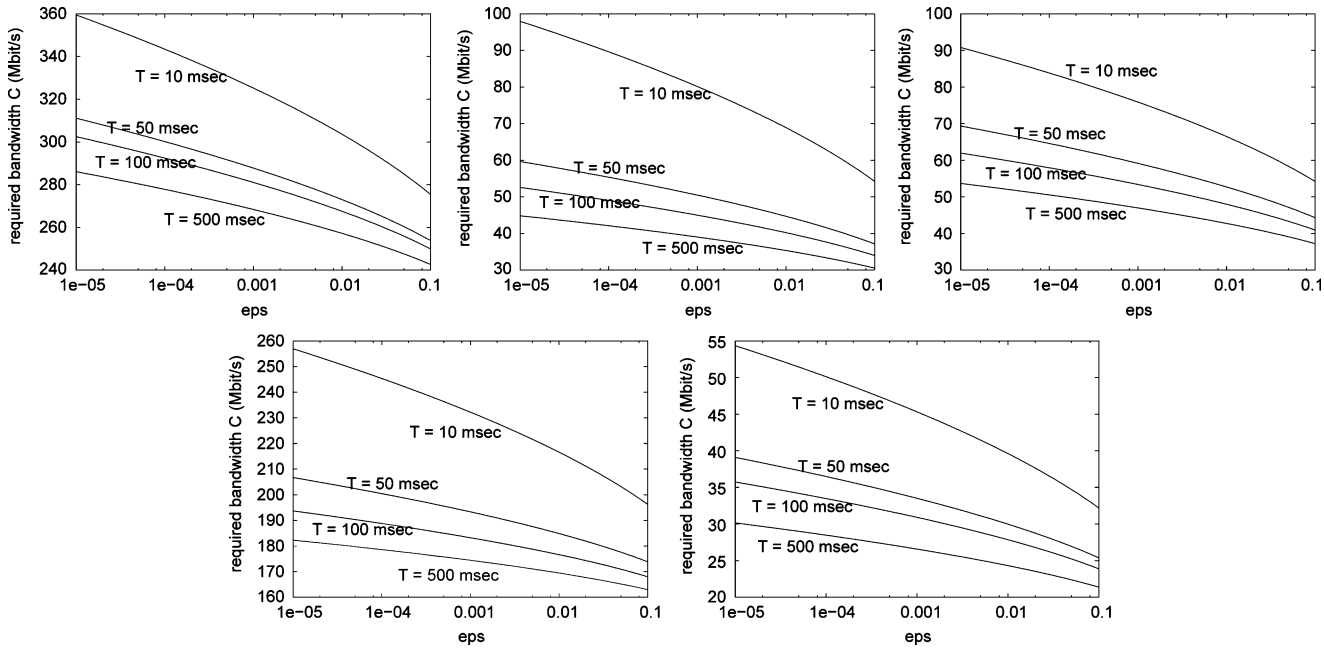

Fig. 13. Required bandwidth for other settings of T and ε for locations {U, R, C, A, S}.

We perform the inversion, and we insert the resulting variance into (4) and see how well this predicts the required bandwidth.

First, we estimate the average traffic rate M and variance $V(T)$ directly from the offered traffic stream. Also, we 'replayed' our traces by inserting them into a virtual queue with capacity c_q (emulated in a Perl script) so that we can use the inversion algorithm to estimate $V(T)$. We denote this directly estimated variance by $V_{\mathsf{direct}}(T)$ and the estimate resulting from inversion by $V_{\mathsf{inversion}}(T)$. Again, we emphasize that the direct estimator $V_{\mathsf{direct}}(T)$ may be infeasible in operational environments, as T is likely chosen to be rather small. In our case studies, we have chosen to set the buffer occupancy sampling interval to 1 s, which ensures that we have a sufficient number of snapshots to reliably estimate the BCD (see Section V). Furthermore, we set T, the time-scale for which we aim to determine the variance, to 100 ms.

The remaining parameter to be set is c_q, the queue's service rate. Clearly, when c_q is chosen too small, say $\mathrm{c}_q < M$, the system is not stable in that it cannot serve all the traffic offered. Hence, we should have $\mathrm{c}_q \geq M$. On the other hand, if c_q is much larger than M, then the queue's occupancy will, obviously, be zero at most observation times. This would lead to an unreliable estimation of the BCD, so c_q should not be too large.

We have performed hundreds of experiments using all our traces to see if there is a general guideline for choosing c_q (for instance, $\mathrm{c}_q = c \cdot M$, where c is some constant) which ultimately leads to an accurate approximation of $V_{\mathsf{inversion}}(T)$; see [26, Appendix B]. It turns out that *per location* c is fairly constant. Hence, for any location, it suffices to determine c on the basis of a few traces; then, this value can be used when approximating $V_{\mathsf{inversion}}(T)$ for the other traces at that location. We also remark that it turns out that choosing c anywhere between 1 and 2 always gives reasonable results; i.e., $V_{\mathsf{inversion}}(T)$ and $V_{\mathsf{direct}}(T)$ are in most cases within 5%–20% of each other.

Furthermore, we stress that the impact of an estimation error in $V_{\mathsf{inversion}}(T)$ on our ultimate goal of bandwidth dimensioning is somewhat mitigated. In (4), first the square-root of the variance is taken, and then it is added to some average rate M. To study the accuracy of the estimation, we introduce

$$\nu := \frac{V_{\mathsf{inversion}}(T)}{V_{\mathsf{direct}}(T)}.$$

As we are interested in using the variance estimates for dimensioning purposes, we compare the required bandwidth $\mathrm{c}_{\mathsf{direct}}$ (i.e., bandwidth estimated using the directly estimated variance) and $\mathrm{c}_{\mathsf{inversion}}$ (i.e., bandwidth estimated using the inversion approach). Recall that $\mathrm{c}_{\mathsf{direct}}$ is an accurate estimation of the required bandwidth, as we saw earlier in the present subsection. We also compare $\mathrm{c}_{\mathsf{inversion}}$ and the 'empirical' minimally required bandwidth $\mathrm{c}_{\mathsf{empirical}}$

$$\mathrm{c}_{\mathsf{empirical}} := \min\left\{\mathrm{c} \;:\; \frac{\#\{A_i \mid A_i > \mathrm{c}T\}}{n} \leq \varepsilon\right\}$$

to be interpreted as the minimum bandwidth that provides the trace with the desired performance. Also, we introduce Δ as an indicator of the quality of the estimation of the required bandwidth through the inversion approach, with respect to both $\mathrm{c}_{\mathsf{direct}}$ and $\mathrm{c}_{\mathsf{empirical}}$

$$\Delta_{\mathsf{var}} := \frac{\mathrm{c}_{\mathsf{inversion}}}{\mathrm{c}_{\mathsf{direct}}} \quad \text{and} \quad \Delta_{\mathsf{cap}} := \frac{\mathrm{c}_{\mathsf{inversion}}}{\mathrm{c}_{\mathsf{empirical}}}.$$

Table II lists the validation results of a number of 'test traces.' it shows that the variances are rather accurately estimated. Table IV compares the estimated required capacity (with $\varepsilon = 0.01$, $T = 100$ ms) computed via both the direct and inversion approaches. Also, the empirically found minimum required bandwidth is tabulated.

As can immediately be seen from the values of Δ_{var}, the required bandwidth capacity as estimated through the inversion approach to estimate the variance is remarkably close to that obtained through the direct approach. On average, the differences are less than 1%. Also, comparison with the empirical minimum

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

TABLE II
VALIDATION RESULTS FOR THE BURSTINESS ESTIMATION METHODOLOGY (M IS IN Mb/s, V IS IN Mb^2; $T = 100$ ms)

trace	M	$V_{\text{direct}}(T)$	$V_{\text{inversion}}(T)$	ν
loc. U #1	207.494	3.773	4.026	1.067
#2	238.773	7.313	7.690	1.052
loc. R #1	18.927	0.492	0.483	0.981
#2	3.253	0.058	0.062	1.062
loc. C #1	23.894	0.633	0.644	1.018
#2	162.404	10.650	12.379	1.162
loc. A #1	147.180	0.939	1.064	1.133
#2	147.984	0.745	0.747	1.003
loc. S #1	14.254	0.200	0.201	1.004
#2	2.890	0.023	0.023	1.022

TABLE III
VALIDATION RESULTS FOR THE BURSTINESS ESTIMATION METHODOLOGY (OVERALL RESULTS)—UPPER TRACES; LOWER WITH $\gamma > 0.9$; $T = 100$ ms AND $\varepsilon = 0.01$

location	avg Δ_{var}	stderr Δ_{var}	avg Δ_{cap}	stderr Δ_{cap}
U	1.00	0.01	1.01	0.07
R	0.96	0.10	0.90	0.19
C	1.00	0.03	1.04	0.11
A	1.00	0.01	1.04	0.02
S	1.00	0.03	0.99	0.10

location	avg Δ_{var}	stderr Δ_{var}	avg Δ_{cap}	stderr Δ_{cap}
U	1.00	0.01	1.03	0.06
R	1.00	0.02	1.00	0.10
C	1.00	0.02	1.05	0.08
A	1.00	0.01	1.04	0.01
S	1.00	0.01	1.01	0.05

required bandwidth, through Δ_{cap}, shows that the use of the inversion procedure leads to estimates for the required bandwidth that are, remarkably, on average less than 4% off. Comparing the respective values for ν and Δ_{var} in Tables II and IV, one observes that an estimation error in $V(T)$ indeed has only limited impact on the error in C.

Now, we have seen the impact of using indirectly estimated variances for a number of test traces (i.e., through the inversion approach). It remains to assess the overall accuracy of our dimensioning approach. We have computed the Δ values as described above for *all* our traces at every location; see the upper part of Table III. We have tabulated the average values of Δ_{var} and Δ_{cap} as well as their standard error terms. The required bandwidth estimations are remarkably accurate. In the lower part of Table III, we have tabulated the same metrics but only used the traces that are "fairly Gaussian," in that their linear correlation coefficient is above 0.9. This improves the results even further—leading to the conclusion that errors in the required bandwidth estimation using the inversion approach to estimate the variance are primarily caused by non-Gaussianity of the offered traffic.

Remark: So far, we concentrated on dimensioning under the link transparency criterion. However, as indicated above, one of the major advantages is that our inversion procedure yields the entire variance curve (up to some threshold). Therefore, it also enables us to estimate C_2 in (5), i.e., to find the link rate such that the probability of exceeding B is below ε. As an example, Fig. 12 compares for a trace at location R, as a function of the buffer size B, the empirically determined C_2 with the one obtained by estimating $V(\cdot)$ by means of our inversion procedure;

TABLE IV
VALIDATION RESULTS FOR THE BURSTINESS ESTIMATION METHODOLOGY (CONTINUED) (C IS IN Mb/s; $T = 100$ ms. AND $\varepsilon = 0.01$)

trace	$C_{\text{empirical}}$	C_{direct}	$C_{\text{inversion}}$	Δ_{var}	Δ_{cap}
loc. U #1	258.398	266.440	268.385	1.007	1.039
#2	302.663	320.842	322.934	1.007	1.067
loc. R #1	37.653	40.221	40.020	0.995	1.063
#2	10.452	10.568	10.793	1.021	1.033
loc. C #1	44.784	48.033	48.250	1.005	1.077
#2	265.087	261.444	269.182	1.030	1.015
loc. A #1	171.191	176.588	178.480	1.011	1.043
#2	168.005	174.178	174.218	1.000	1.037
loc. S #1	27.894	27.843	27.873	1.001	0.999
#2	7.674	7.482	7.532	1.007	0.981

we take $\varepsilon = 0.01$. The performance of our dimensioning procedure under (5) is very similar to that under (4), as reported. □

C. Multilink Scenarios; Practical Guidelines

We now describe how our dimensioning procedure extends to a network setting. Let there be I routes, and let r_i be the set of links on route i, for $i = 1, \ldots, I$. Allow any route to have specific end-to-end performance requirements. On route r_i, the performance criterion is that, in self-evident notation

$$\mathbb{P}\left(\exists \ell \in r_i : A_\ell(T_i) > \mathrm{C}_\ell T_i\right) \leq \varepsilon_i.$$

Observe that the T_i and ε_i can be chosen route-specific. (Notice that this is the "network variant" of the transparency criterion $\mathbb{P}(A(T) \geq \mathrm{C}T) \leq \varepsilon$; in the same way, one can define the network variant of the criterion $\mathbb{P}(Q > \mathrm{B}) \leq \varepsilon$.) Similar to Fraleigh *et al.* [14, Sec. III], the above probability can be approximated by

$$p_i := \sum_{\ell \in r_i} \exp\left(-\frac{1}{2}\frac{(\mathrm{C}_\ell - M_\ell)^2}{V_\ell(T_i)}\right)$$

relying on well-studied decomposition properties [19], [32] and the standard approximation $1 - \prod_{i=1}^{m}(1 - p_i) \approx \sum_{i=1}^{m} p_i$. We thus find that solving the capacity allocation problem requires knowledge of the M_ℓ (i.e., the per-link mean) and the $V_\ell(T)$ (i.e., the per-link variance; required for the time-scale $T = T_i$ if link ℓ is used by route r_i, for any $i = 1, \ldots, I$). Importantly, these can be estimated by performing measurements on a per-link basis; the accuracy of the resulting estimates was already described in detail in Section VI-B. The capacity allocation problem then reduces to solving a fairly standard optimization problem; we could, for instance, minimize the total bandwidth required $\sum_\ell \mathrm{C}_\ell$ under $p_i \leq \varepsilon_i$, for $i = 1, \ldots, I$ (where it is recalled that the p_i depends on the link rates).

The procedure described above requires estimates of the mean and variance on every link. Alternatively, in the situation that routing information is available, one could infer these from the mean and variance at the ingress. Such an approach could reduce substantially the measurement effort needed. An alternative dimensioning approach for the multilink setting (with emphasis on meeting resilience requirements) can be found in [27].

We advise to use our link dimensioning methodology on a periodic basis (and to adapt the link capacity when needed) so as to prevent links becoming (systematically) congested. If, despite these periodic checks, the link becomes congested, it is clear that traffic measurements do not reflect well the users' demand.

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

In the first place, this is due to the fact that on a congested link, TCP's closed loop-control will strongly affect the traffic pattern. In addition to this, in case of congestion also, *the users* will adapt their behavior: Long response times could frustrate them, and as a result, their demand is typically lower than in a congestion-free network.

As a consequence, on (systematically) congested links, resource dimensioning should not be (solely) based on traffic measurements, and an alternative approach needs to be followed. Such an approach should be, in the terminology of [24], "user-oriented." This means that one does not model traffic aggregates (as we did in the present paper) but rather the random dynamics of the individual flows. This flow-level traffic characterization could then be done as follows, cf. [3]: Identify a number of user classes, estimate (per user class) the traffic characteristics, and characterize the dynamics of the number of simultaneous flows (of each user class) at the link under consideration. To this end, one could use by measurements at the access of the network (and routing information). Having estimated such a flow-level traffic model, one could then rely on flow-level queueing models [6] to determine the capacity needed. As an aside, we remark that this flow-level traffic estimation is usually substantially more involved than estimating the parameters of a Gaussian model (see [3] and [24]) and, therefore, we would propose to follow this approach only for congested links.

Our approach relies on the assumption that traffic characteristics are hardly affected by TCP's feedback loop, and therefore, a next question is up to what loads this property is valid. To assess this issue, we set up NS-2 simulations in which traffic streams, roughly modeled in accordance with our measurement locations (in terms of flow sizes, access rates, etc.), feed into a link of capacity C, and estimated $V(T)$. Then, we repeated the experiment for different values of C. In all scenarios, we find that up to an average link load of approximately 70%–75% of C, the standard deviation $\sqrt{V(T)}$ remains more or less constant (that is, less than 4% off; in most cases, substantially less). For higher values of the average link load, TCP does have significant impact in that it has a noticeable effect on the flows' transmission rates (in fact, due to this "throttling," the variance tends to decrease).

Our method was developed for a network that is already in operation, but it can also be used when an increase of the traffic volume is anticipated, as follows. Consider, for instance, the situation that at some location the number of users increases from N_1 to N_2. The properties of the normal distribution entail that both the mean and the variance of the traffic stream increase by a factor $f := (N_1 + N_2)/N_1$; the new mean and variance become fM and $fV(\cdot)$, for $f > 1$. (Interestingly, this does *not* imply that the required bandwidth is also multiplied by f. Applying dimensioning formula (4), we see that the new link rate becomes

$$fM + \sqrt{f} \cdot \frac{\delta}{T} \cdot \sqrt{V(T)} < f\left(M + \frac{\delta}{T} \cdot \sqrt{V(T)}\right).$$

Due to statistical multiplexing, the required bandwidth grows less than proportional.)

This approach relies on the assumption that the user group added is, approximately, "statistically identical" to the user group that was already present. We verified this "homogeneity property" at various locations at various time-scales; it was also extensively validated in our previous work [5, Sec. IV].

VII. Discussion and Concluding Remarks

This paper provides a framework for link dimensioning. We first derived generic dimensioning formulas, which require knowledge of the variance of the input traffic. Then, we proposed an efficient variance estimation technique. This so-called inversion method was extensively tested in experiments with real network traffic from various representative networking environments. These tests showed that the resulting estimated link rates are remarkably accurate.

In the remainder of this section, we discuss a number of directions for further research and reflect on alternative approaches.

1) Extensions, Further Research: A prerequisite for our method is the *performance criterion*; when adopting the link transparency criterion, one needs to specify appropriate values for the parameters T and ε. These values are dictated by the so-called *perceived quality-of-service*, i.e., the performance as perceived by users. Clearly, any application has other requirements, but these are also not necessarily uniform among users of the same application—performance is a subjective issue. For traditional services (such as voice), the mapping from (subjective) perceived performance to (objective) performance parameters (such as T and ε) has been intensively studied; for more advanced applications (for instance interactive web applications), this is still an open issue.

Often, dimensioning issues cannot be addressed without taking into account pricing aspects. Put differently, the utility a network user assigns to the service he is offered depends not only on the performance level but also on the price he has to pay for it; in fact, the combination of performance and price will determine whether a potential user is willing to subscribe. From this perspective, it is not the providers' task to choose the minimum bandwidth such that a prespecified performance target is met, but rather to choose bandwidth and prices such that some objective function (for instance revenues, or profits) is optimized.

2) Other Approaches: The purpose of our inversion method is to retrieve the essential traffic characteristics at low measurement costs. We remark that several other "cheap" (i.e., with low measurement effort) methods have been proposed in the literature. We now discuss some of these.

If one cannot assume Gaussianity, one could still use approximation (2) and estimate the mgf $\mathbb{E}e^{\vartheta A(-t,0)}$ (as a function of both t and ϑ) from traffic measurements. There are several papers on its statistical aspects; see, for instance, [16]. As we have seen, the formula simplifies considerably when assuming Gaussian traffic, as then estimating the mgf reduces to estimating M and $V(\cdot)$.

The paper by Duffield *et al.* [12] presents a procedure to accurately estimate the *asymptotic cumulant function* $\Lambda(\vartheta) := \lim_{t\to\infty} t^{-1} \cdot \log \mathbb{E}e^{\vartheta A(t)}$ from traffic measurements. Knowledge of this function is useful because, under some assumptions on the traffic arrival process, it holds that $\mathbb{P}(Q > \mathrm{B}) \approx \exp(-\vartheta * \mathrm{B})$, for B large, where $\vartheta *$ solves the equation $\Lambda(\vartheta) = \mathrm{C}\vartheta$. The crucial assumption, however, is that the arrival process be short-range dependent; otherwise, one cannot be sure that the cumulant exists. Think of fBm, for which $\mathbb{E}\exp(\vartheta A(t))$ is of the form $\exp(M\vartheta t + \frac{1}{2}\vartheta^2 t^{2H})$, so that, for $H \in (\frac{1}{2}, 1)$

$$\lim_{t\to\infty} \frac{1}{t} \log \mathbb{E}e^{\vartheta A(t)} = \lim_{t\to\infty} \left(\vartheta M + \frac{1}{2}\vartheta^2 t^{2H-1}\right) = \infty.$$

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.

The fact that this method cannot be used for long-range dependent traffic makes its use for dimensioning purposes limited. We remark that a crucial difference with our approach is that [12] and [16] measure *traffic*, whereas we propose to measure (or better, to sample) the *buffer content*.

Another related study is by Kesidis *et al.* [8]. As in our method, their approach relies on the estimation of the buffer content distribution $\log \mathbb{P}(Q > \mathrm{B})$. Under the assumption of short-range dependent input, $\log \mathbb{P}(Q > \mathrm{B})$ is linear for large B (with slope $-\vartheta*$). Having estimated $\vartheta*$, the probability of overflow over higher buffer levels can be estimated by extrapolating $\log \mathbb{P}(Q > \mathrm{B})$ linearly. For long-range dependent traffic, $\log \mathbb{P}(Q > \mathrm{B})$ is not linear (see Fig. 7), and hence, this method cannot be used.

Another approach to minimize the measurement effort was presented in [5]. There, traffic is assumed to be Gaussian, but with the correlation structure of the $\mathrm{M}/\mathrm{G}/\infty$ input model. Interestingly, under the link transparency criterion, the required bandwidth simplifies to $\mathrm{C} = M + \alpha\sqrt{M}$, where α depends on the performance criterion (i.e., T and ε) and on the characteristics of a single flow (i.e., the distribution of the flow duration B and the traffic rate r) but not on the traffic arrival rate λ. This property enables a simple estimate of the additionally required bandwidth if, in a future scenario, traffic growth is mainly due to a change in λ (e.g., due to growth of the number of subscribers) and not due to changes in user behavior.

REFERENCES

[1] R. Addie, P. Mannersalo, and I. Norros, "Most probable paths and performance formulae for buffers with Gaussian input traffic," *Eur. Trans. Telecommun.*, vol. 13, pp. 183–196, 2002.

[2] C. Barakat, P. Thiran, G. Iannaccone, C. Diot, and P. Owezarski, "Modeling Internet backbone traffic at the flow level," *IEEE Trans. Signal Process.*, vol. 51, no. 8, pp. 2111–2114, Aug. 2003.

[3] N. Ben Azzouna, F. Clérot, C. Fricker, and F. Guillemin, "A flow-based approach to modeling ADSL traffic on an IP backbone link," in *Ann. Telecommun.*, 2004.

[4] J. Beran, *Statistics for Long-Memory Processes*. London, U.K.: Chapman & Hall/CRC, 1994.

[5] H. van den Berg, M. Mandjes, R. van de Meent, A. Pras, F. Roijers, and P. Venemans, "QoS-aware bandwidth provisioning of IP links," *Comput. Netw.*, vol. 50, pp. 631–647, 2006.

[6] T. Bonald, P. Olivier, and J. Roberts, "Dimensioning high speed IP access networks," in *Proc. ITC 18*, Berlin, Germany, 2003, pp. 241–251.

[7] D. Botvich and N. Duffield, "Large deviations, the shape of the loss curve, and economies of scale in large multiplexers," *Queueing Syst.*, vol. 20, pp. 293–320, 1995.

[8] C. Courcoubetis, G. Kesidis, A. Ridder, J. Walrand, and R. Weber, "Admission control and routing in ATM networks using inferences from measured buffered occupancy," *IEEE Trans. Commun.*, vol. 43, no. 2–4, pp. 1778–1784, Feb.–Apr. 1995.

[9] M. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic. Evidence and possible causes," *IEEE/ACM Trans. Netw.*, vol. 6, pp. 835–846, Dec. 1997.

[10] K. Debicki and Z. Palmowski, "Heavy-traffic Gaussian asymptotics of on-off fluid model," *Queueing Syst.*, vol. 33, pp. 327–338, 1999.

[11] T. Dieker, "Fractional Brownian motion simulator," 2002 [Online]. Available: http://homepages.cwi.nl/~ton/fbm/index.html

[12] N. Duffield, J. Lewis, N. O'Connell, R. Russell, and F. Toomey, "Entropy of ATM traffic streams: A tool for estimating QoS parameters," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 6, pp. 981–990, Aug. 1995.

[13] A. Erramilli, O. Narayan, and W. Willinger, "Experimental queueing analysis with long-range dependent packet traffic," *IEEE/ACM Trans. Netw.*, vol. 4, no. 2, pp. 209–223, Apr. 1996.

[14] C. Fraleigh, F. Tobagi, and C. Diot, "Provisioning IP backbone networks to support latency sensitive traffic," in *Proc. IEEE INFOCOM*, San Francisco, CA, 2003, online.

[15] A. Ganesh, N. O'Connell, and D. Wischik, *Big Queues*. Berlin, Germany: Springer-Verlag, 2004.

[16] L. Györfi, A. Rácz, K. Duffy, J. Lewis, and F. Toomey, "Distribution-free confidence intervals for measurements of effective bandwidth," *J. Appl. Probability*, vol. 37, pp. 224–235, 2000.

[17] I. Juva, R. Susitaival, M. Peuhkuri, and S. Aalto, "Traffic characterization for traffic engineering purposes: Analysis of Funet data," in *Proc. 1st EURO-NGI Conf.*, Rome, Italy, 2005.

[18] F. Kelly, "Notes on effective bandwidths," in *Stochastic Networks: Theory and Applications*, F. P. Kelly, S. Zachary, and I. B. Ziedins, Eds. London, U.K.: Oxford Univ. Press, 1996, pp. 141–168.

[19] L. Kleinrock, *Queueing Systems, Volume II: Computer Applications*. New York: Wiley Interscience, 1976.

[20] J. Kilpi and I. Norros, "Testing the Gaussian approximation of aggregate traffic," in *Proc. Internet Meas. Workshop*, Marseille, France, 2002 [Online]. Available: http://www.vtt.fi/tte/rd/traffic-theory/papers/

[21] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of ethernet traffic (extended version)," *IEEE/ACM Trans. Netw.*, vol. 2, no. 1, pp. 1–15, Feb. 1994.

[22] M. Mandjes and R. van de Meent, R. Boutaba, K. Almeroth, R. Puigjaner, S. Shen, and J. Black, Eds., "Inferring traffic burstiness by sampling the buffer occupancy," in *Proc. 4th Int. IFIP-TC6 Netw. Conf.*, Waterloo, ON, Canada, 2005, LNCS Series, 3462, pp. 303–315.

[23] M. Mandjes, I. Saniee, and A. Stolyar, "Load characterization and load anomaly detection for voice over IP traffic," *IEEE Trans. Neural Netw.*, vol. 16, no. 5, pp. 1019–1028, Sep. 2005.

[24] R. van de Meent and M. Mandjes, "Evaluation of 'user-oriented' and 'black box' traffic models for link provisioning," in *Proc. 1st EURO-NGI Conf.*, Rome, Italy, 2005.

[25] R. van de Meent, M. Mandjes, and A. Pras, "Gaussian traffic everywhere?," in *IEEE Int. Conf. Commun.*, Istanbul, Turkey, 2006, vol. 2, pp. 573–578.

[26] R. van de Meent, "Network link dimensioning: A measurement & modeling based approach," Ph.D. dissertation, Univ. Twente, Enschede, The Netherlands, 2006.

[27] M. Menth, R. Martin, and J. Charzinski, "Capacity overprovisioning for networks with resilience requirements," in *Proc. SIGCOMM*, 2006, pp. 87–98.

[28] I. Norros, "A storage model with self-similar input," *Queueing Syst.*, vol. 16, pp. 387–396, 1994.

[29] I. Norros, "On the use of fractional Brownian motion in the theory of connectionless networks," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 6, pp. 953–962, Aug. 1995.

[30] V. Paxson and S. Floyd, "Wide-area traffic: The failure of Poisson modeling," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, pp. 226–244, Jun. 1995.

[31] G. A. Seres, Szlávik, J. Zátonyi, and J. Bíró, "Alternative admission rules based on the many-sources asymptotics," in *Proc. 7th ISCC*, 2002, pp. 995–1000.

[32] D. Wischik, "The output of a switch, or, effective bandwidths for networks," *Queueing Syst.*, vol. 32, pp. 383–396, 1999.

Michel Mandjes received the M.Sc. degree in both mathematics and econometrics and the Ph.D. degree from the Vrije Universiteit (VU), Amsterdam, The Netherlands.

He has worked as a Member of Technical Staff at KPN Research, Leidschendam, The Netherlands, and Bell Laboratories/Lucent Technologies, Murray Hill, NJ; a part-time Full Professor at the University of Twente, Enschede, The Netherlands; and Department Head at CWI, Amsterdam, The Netherlands. He currently holds a full professorship at the University of Amsterdam, Amsterdam, The Netherlands. His research interests include performance analysis of communication networks, queueing theory, Gaussian traffic models, traffic management and control, and pricing in multiservice networks.

Remco van de Meent received the M.Sc. degree in computer science and the Ph.D. degree from the University of Twente, Enschede, The Netherlands, in 2001 and 2006, respectively.

He is currently working as a Researcher/Designer at Vodafone NL, Maastricht, The Netherlands. His research interests include network dimensioning, traffic modeling, network security, and network management.

Authorized licensed use limited to: UVA Universiteitsbibliotheek SZ. Downloaded on April 19,2010 at 13:40:24 UTC from IEEE Xplore. Restrictions apply.