Burstiness predictions based on rough network traffic measurements

R. van de Meent[⋆,†], A. Pras[†], M.R.H. Mandjes[†,⊤], J.L. van den Berg[†,‡],
F. Roijers[‡], L.J.M. Nieuwenhuis[†], P.H.A. Venemans[‡]
[⋆]*Corresponding author (Email: r.vandemeent@utwente.nl)*
[†]*University of Twente, PO Box 217, 7500 AE Enschede, The Netherlands*
[⊤]*CWI, Amsterdam, The Netherlands* [‡]*TNO Telecom, Delft, The Netherlands*

Abstract. To dimension network links, such that they will not become QoS bottlenecks, the peak rate on these links should be known. To measure these peaks on sufficiently small time scales, special measurement tools are needed. Such tools can be quite expensive and complex. Therefore network operators often rely on more cheap, standard tools, like MRTG, which were designed to measure average traffic rates (m) on time scales such as 5 minutes. For estimating the peak traffic rate (p), operators often use simple rules, such as $p = \alpha \cdot m$. In this paper we describe measurements that we have performed to investigate how well this rule describes the relation between peak and average traffic rate. In addition, we propose some more advanced rules, and compare these to the simple rule mentioned above. The analyses of our measurements, which have been performed on different kinds of networks, show that our advanced rules more adequately describe the relation between peak and average traffic rate.

Keywords: *traffic measurements, link dimensioning, bandwidth provisioning*

1 Introduction

To achieve a sufficiently good Quality of Service (QoS) level, operators' network links have to be dimensioned such that traffic bursts on small time scales are transmitted without experiencing substantial congestion. Hence, to be able to prevent such overload situations, it is of crucial importance to know (an estimation of) the peak traffic rate. To determine the peak traffic rate, detailed (i.e., on a small time scale) measurements should be performed, requiring considerable effort (in terms of resources and cost). Long-term, e.g., 5 minutes, average traffic rates however, can be estimated easily by using standard tools like the Multi Router Traffic Grapher (MRTG) [2]. Thus it would be useful to have a methodology that shows the relation between the average and peak traffic rate, i.e., without the need to constantly perform detailed measurements. Ideally, one would like to have a fixed mathematical rule that predicts the peak traffic rate on a network link, using average traffic rate information.

Contribution. We propose and evaluate (mathematical) rules that determine the (statistical) relation between the average network traffic rate, as can be measured using MRTG on a time scale of 5 minutes, and the peak traffic rate on, e.g., a time scale of 1 second. Because of our ultimate goal, i.e., configuration management and link dimensioning in particular, we chose to determine rules that are conservative in terms of relating average to peak network traffic rates.

Approach. We follow an empirical approach to find the relation between average and peak traffic rates. Therefore we have performed (detailed) traffic measurements on existing networks. We introduce these measurements in Section 2. Our analyses start in Section 3, with determining to what extent a simple rule like "50% overdimensioning is required to handle traffic peaks" suitably expresses the relation between average and peak traffic rates. This (linear) rule can be written as $p = \alpha \cdot m$, where m is the average traffic rate, and p the peak traffic rate. In Section 4 we subsequently investigate a number of alternative, more advanced rules to express the relation between average and peak traffic rates. Section 5 concludes.

2 Measurements

The measurements in this study are performed on so-called "uplinks" connecting various access networks to core ("backbone") networks. See [3] for an overview of our measurement setup, equipment and software. The three uplinks that have been measured in this study, can be characterized as follows: #1 is the 1 Gbit/s link connecting an organization of about 200 people, each having a FastEthernet access link, to the backbone of the Dutch research and educational network. The uplink of network #2 carries the aggregated traffic of about 1000 users; their access network is similar to #1, as is the uplink's capacity. To verify our findings in another (common) network infrastructure, i.e., ADSL access, we have performed measurements on a number of uplinks of DSLAMs to a commercial ADSL operator's backbone network. We refer to these DSLAM uplinks as network #3.

Our general approach is that we measure the throughput per second, within a 5 minute interval. From the resulting 300 throughput rates, we calculate the 5 minute average (referred to as m). In addition, we determine p, which denotes the 99th percentile of the 1 second traffic rates. This measurement procedure is repeated multiple times for each of the networks. Note that, when keeping (efficient) link dimensioning in mind, it is not illogical to omit a few (exceptionally) high measurement values. The choices of time scale for the (detailed) measurements (T), as well as the percentile ($1 - \epsilon$) that is used as the basis to determine the peak traffic rate, will be application-dependent. Our choices for $T = 1$ second and $\epsilon = 0.01$ are motivated by our expectation that these values corresponds to time scales and QoS objectives relevant to users browsing the WWW. Although for other applications the values for T and/or ϵ may change, the methodology presented in this paper can remain. Note that smaller values for T and ϵ, will lead to higher values of p [3].

3 A Simple Rule

In this section we investigate the effectiveness of the simplest approach to relating average to peak traffic rates: "the peak traffic rate is, e.g., 50% higher than the average traffic rate". This corresponds to the linear relation $p = \alpha \cdot m$, in which the parameter α is $1\frac{1}{2}$.

First we present the rough measurement data, and discuss the need to omit a certain (small) part of this data in order to get rid of "outliers". Next, we follow a numerical approach to find a linear rule to describe the relation between m and p, which fits the measurement data, and we introduce a criterion to objectively measure the effectiveness of that rule.

Measurement results. The measurements on network #1 provide numerous combinations of mean and peak traffic rates: (m, p)-tuples. We plot all the (m, p)-tuples in a single graph, as shown in Figure 1. The graph shows a dense cloud, and a group of some 20–30 tuples that fall outside this cloud. The latter group can be divided into two subgroups: the tuples on the right-hand side of the cloud, and the tuples above the cloud. The reason that there are only few tuples right of the cloud is caused by the simple fact that there are only a few instances where the 5 minute average throughput rate is that high. The tuples above the cloud are caused by measurements in which there are a few seconds with a large throughput rate, which is possible by relatively small flows that have a large throughput rate. Such flows are possible because of the high access link speed.

Since the context of our study is link dimensioning, we are not so much interested in finding a rule that, for given m, estimates the *average* p, but rather the *maximum* p. This is because a rule for average p underestimates the peak traffic rate in approximately half of all cases. Once we know, for each m, the corresponding maximum p, we can draw a line through these (m, p)-tuples. In principle all tuples will be below or on this line. In this section, we focus on the rule $p = \alpha \cdot m$. If we would draw such a line in Figure 1, the line would be a straight line

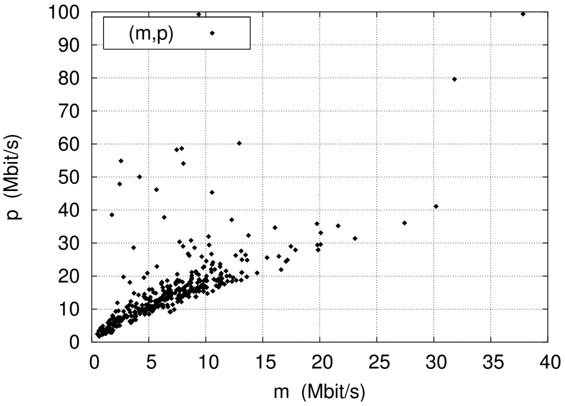

Figure 1: m v. p for network #1, outliers not removed

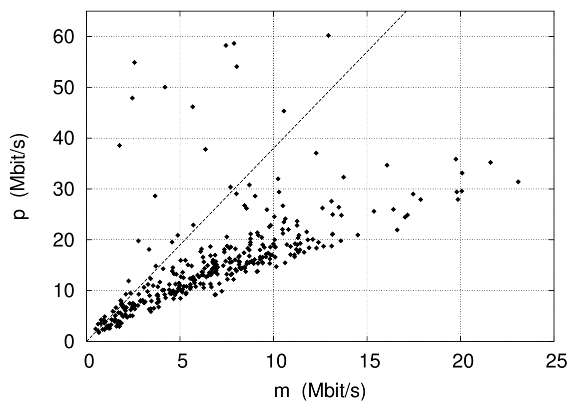

Figure 2: m v. p for network #1, fit: $\hat{p} = 3.80 \cdot m$

through the root $(0, 0)$. The line would have a steep slope, which is determined by a small number of "outliers": the points at the left of the figure, above the cloud. Since we do not want our rules to be affected by these outliers, we change the requirement that a rule should estimate the maximum p for given m, into *90% of the measured* (m, p)*-tuples must be below the line that is inferred from the rule*. Basically, this means that we do not want to honour 100% of the resource claims, but only 90%. Note, however, that this does not necessarily mean that the 10% "outliers" receive a bad QoS. From experiments not described in this paper, we learned that other choices for "outlier filtering", e.g., "95% of the tuples should fall below the line", do not significantly change the conclusions of this paper.

In the remainder of this paper, we will present the various rules while considering 10% of all (m, p)-tuples as outliers. As for notation, we will use $\hat{p}$ to denote the *estimation* of the peak traffic rate for given average traffic rate m; m and p refer to individual measurements.

Linear relation. The simplest relation between average and peak traffic rates is a linear relation of the form:

Rule 1: $\hat{p} = \alpha \cdot m$

Thus we want to approximate the cloud of (m, p)-tuples with a straight line through the root $(0, 0)$. As argued in 3.1, we want to find a fit that gives sufficiently large values of $\hat{p}$ for 90% of the measurement points. There is no standard mathematical approach to find this fit. We therefore determine the value of α numerically, by starting at its minimum value of 1 (corresponding to $\hat{p} = m$), and then slowly increasing the slope of the rule until 90% of all (m, p)-tuples are below or on this line.

We have determined, see Figure 2, that for network #1 the best fit of the form of rule 1 is $\hat{p} = 3.80 \cdot m$. An important observation is that for larger values of m, the line that has been found overshoots the cloud significantly, i.e., there is a large difference between the measured p and the estimation $\hat{p}$. The figure also shows that most outliers are found to stem from relative small values of m. Note that for presentation purposes, tuples with $m > 25$ are not shown in this figure; they have, however, been taken into account in the calculations that lead to the fit.

In order to verify the results from the measurements on network #1, the same procedure has been applied to networks #2 and #3, see Figure 3 and Table 1. For network #2, it looks like that there are hardly any outliers. This is due to the fact that the cloud is very dense for small values of m. Note that the α required to meet the requirement of "90% of all (measured) peaks should be below the line" is considerably smaller in networks #2 (i.e., 2.27) and #3 (i.e., 1.61), compared to network #1. This observation can be explained by the fact that (on average) the

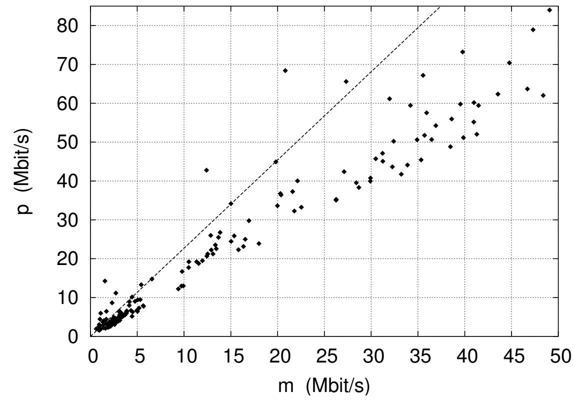

Figure 3: m v. p for network #2, fit: $\hat{p} = 2.27 \cdot m$

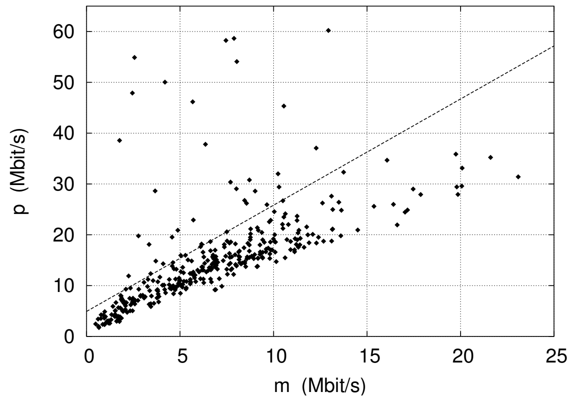

Figure 4: m v. p for network #1, fit: $\hat{p} = 2.09 \cdot m + 4.92$

access link rates are smaller compared to the capacity of the measured link, thereby limiting the impact that a small number of users can have on the aggregate traffic, which implies that the average peak-to-mean ratio also decreases.

A criterion for goodness-of-fit. In order to objectively judge how effectively a rule fits the cloud, we introduce the following measure, for which we coin the term *average relative overshooting* Φ:

$$\Phi : \quad \frac{1}{\#p_i} \sum_i \frac{\hat{p}_i - p_i}{p_i} \cdot 100\% \;, \quad \forall i : p_i \leq \hat{p}_i$$

This formula determines for all (m, p)-tuples that are below the line, the relative difference between the measured value of p, and the calculated $\hat{p}$. The resulting sum is then averaged over all tuples. The lower the sum, the lower the average relative overshooting, and thus (by definition) the better the fit. In this study, we chose to ignore the measured values of p that are above the line. The reason for this choice that we want to be as accurate as possible for that part of the cloud that meets our requirement of "90% of all (measured) peaks should be below the line". Hence, in this study, it does not matter whether an ignored tuple is just above, or multiple factors above the line. In Table 1, an overview is given of the average relative overshooting for both rule 1 as well as the more advanced rules as will be described in the next section.

4 Advanced Rules

The main problem with rule 1 is that a linear relation of the form $\hat{p} = \alpha \cdot m$ overshoots the measured peaks for larger average traffic rates. In this section we discuss various alternative rules to tackle this problem. The initial approach that we follow is that we add a second parameter to the formula. Generally spoken, the more parameters, the better the fit for a cloud of measurement points can be.

Linear relation plus a constant. The simplest solution to solve the overshooting problem for larger m, is that of decreasing the slope of the line. In order to have the rule to remain applicable for smaller m as well, we need to add a constant, γ, as parameter to the linear relation:

Rule 2: $\quad \hat{p} = \alpha \cdot m + \gamma$

When applied to the measurements from network #1, we find that we can best fit the cloud with: $\hat{p} = 2.09 \cdot m + 4.92$, see Figure 4. The graph shows that we do not "overshoot" as much as without the constant γ, an observation that is confirmed when we calculate the average relative overshooting: 52%, instead of the 85% obtained with rule 1. A drawback is that rule 2 does not

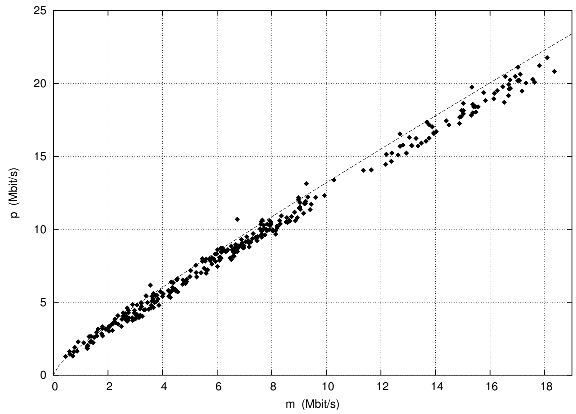

Figure 5: m v. p for network #3, fit: $\hat{p} = m + 1.01\sqrt{m}$

Network	*Rule*	*Fit*	Φ
#1	1	$\hat{p} = 3.80 \cdot m$	83 %
#1	2	$\hat{p} = 2.09 \cdot m + 4.92$	52 %
#1	3	$\hat{p} = 5.42 \cdot m^{0.67}$	47 %
#1	4	$\hat{p} = m + 4.55 \cdot m^{0.54}$	48 %
#1	5	$\hat{p} = m + 4.86 \cdot \sqrt{m}$	49 %
#2	1	$\hat{p} = 2.27 \cdot m$	49 %
#2	2	$\hat{p} = 1.81 \cdot m + 1.45$	39 %
#2	3	$\hat{p} = 2.98 \cdot m^{0.85}$	43 %
#2	4	$\hat{p} = m + 1.99 \cdot m^{0.74}$	43 %
#2	5	$\hat{p} = m + 3.66 \cdot \sqrt{m}$	63 %
#3	1	$\hat{p} = 1.61 \cdot m$	24 %
#3	2	$\hat{p} = 1.19 \cdot m + 1.17$	9 %
#3	3	$\hat{p} = 1.87 \cdot m^{0.85}$	8 %
#3	4	$\hat{p} = m + 0.99 \cdot m^{0.51}$	8 %
#3	5	$\hat{p} = m + 1.01 \cdot \sqrt{m}$	8 %

Table 1: Overview of determined rules and corresponding overshooting numbers

hold for $m = 0$: if there is no traffic, the peak traffic rate should be 0, but with $\gamma > 0$, rule 2 yields $\hat{p} > 0$.

After applying rule 2 on networks #2 and #3, we find that also in these cases average relative overshooting improves; see Table 1 for details.

Non-linear relations. Rules 1 and 2 describe a linear relation between m and p. We now investigate the applicability of a non-linear relation, described by the following two rules:

Rule 3: $\hat{p} = \alpha \cdot m^{\beta}$

From the shape of the cloud of (m, p)-tuples it is clear that the exponent β must be smaller than 1. As the average relative overshooting figures in Table 1 show, rule 3 performs better than rule 1, but not always better than rule 2. A fundamental problem with rule 3, however, is that for m (much) larger than the values we have measured, the estimated $\hat{p}$ will be smaller than m, because of $\beta < 1$.

Rule 4 makes use of the fact that the peak traffic rate consists of two parts: the average traffic rate, plus a fraction that resembles the variation in the traffic rate (burstiness), hereby solving the fundamental problem of rule 3:

Rule 4: $\hat{p} = m + \alpha \cdot m^{\beta}$

For network #1, we find the best fit of the form of rule 4 to be $\hat{p} = m + 4.55 \cdot m^{0.54}$. The verification procedure gives for networks #2 and #3 an α of 1.99 and 0.99, and a β of 0.74 and 0.51, respectively. The resulting average relative overshooting numbers show that rule 4 is comparable in this respect to rules 2 and 3.

Simple additive non-linear relation. The values of β that were found while investigating rule 4, suggest that a simple variant, with only one parameter (α) and a fixed value $\beta = \frac{1}{2}$, might also effectively describe the relation between average and peak traffic rates:

Rule 5: $\hat{p} = m + \alpha \cdot \sqrt{m}$

Remarkably, this rule is the same as a rule developed over half a century ago to predict peak loads from the mean usage in trunks of telephony lines, see e.g., [1].

It turns out that, for networks #1 and #3, this rule gives a comparable average relative overshooting as rules 2 to 4 (see Figure 5 and Table 1). However, rule 5 needs only a single parameter. Thus for networks #1 and #3, the decoupling of $\hat{p}$ in the average traffic rate and the variation in traffic rates gives significant gain compared to the other rule with only one parameter, i.e., rule 1. Unfortunately, for network #2, rule 5 gives a worse result in terms of average relative overshooting than the other rules do. This is probably what we could have expected when looking at the values of β that belong to rule 4: for network #2, β (i.e., 0.74) did certainly not suggest a relation between m and $\hat{p}$ to be of the form of rule 5.

Still we may conclude that, regardless of the heterogeneity and burstiness of Internet traffic, in some scenarios the old telephony model works remarkably good. This can probably be explained by the fact that, given sufficient aggregation of traffic sources and limited end-user traffic rates, the scenario is comparable to that of a trunk of telephone circuits.

5 Conclusions

In this paper we have investigated the relation between average traffic rate on a time scale of 5 minutes, and peaks in the traffic rates on the smaller time scale of 1 second. We have empirically developed rules that describe this relation. These rules give insight in peak traffic rates on a time scale of, e.g., 1 second, while only requiring rough network traffic measurements, e.g., on a time scale of 5 minutes.

From our study on three live networks, we conclude that a traditional, linear formula of the form $p = \alpha \cdot m$ (rule 1) is not the best way to describe the relation between average and peak traffic rates. We have investigated various alternatives to this rule, which are slightly more complicated: $\alpha \cdot m + \gamma$ (rule 2), $\alpha \cdot m^{\beta}$ (rule 3), $m + \alpha \cdot m^{\beta}$ (rule 4) and $m + \alpha \cdot \sqrt{m}$ (rule 5). It turns out that all these alternatives show significantly better results than rule 1. The average relative overshooting, which we introduced as a measure to judge the effectiveness of these rules, is significantly less when compared to rule 1. The differences in results between rules 2 to 5 are minor; it is hard to judge which is in every case the best rule.

From an engineering perspective we may conclude that rule 2 gives the best results, as this is a formula that is easy to understand and to reason about. A drawback, however, is that there are two parameters, and hence it requires considerable effort to find the optimal combination of the parameters. From an analytical and modeling perspective, we may conclude that rule 5 most effectively describes the relation between average and peak traffic rates (under certain circumstances, as argued in Section 4). Rule 5 is attractive, since it gives reasonably good results in terms of average relative overshooting, with only a single parameter.

Future work. In future work, we will study which factors may influence the parameters α, β and γ. It is likely that these parameters depend on the chosen values of T and ϵ, and the specific environment: network topology, applications, number of clients, user behavior, etc. The goodness-of-fit criterion may also be subject of further study, e.g., to include business-level aspects. In future work we will also study the mathematical foundations under the formulas described in this paper.

References

[1] A. Jensen. *The life and works of A.K. Erlang*, volume 2, chapter An elucidation of A.K. Erlang's statistical works through the theory of stochastic processes, pages 23–100. Transactions of the Danish Academy of Technical Sciences, 1948.

[2] Tobias Oetiker. MRTG: Multi Router Traffic Grapher, 2003. `http://people.ee.ethz.ch/~oetiker/webtools/mrtg/`.

[3] Remco van de Meent, Aiko Pras, Michel Mandjes, Hans van den Berg, and Lambert Nieuwenhuis. Traffic Measurements for Link Dimensioning: A Case Study. In M. Brunner and A. Keller, editors, *Proceedings of the 14th IFIP/IEEE Workshop on Distributed Systems: Operations and Management (DSOM2003)*, number 2867 in Lecture Notes in Computer Science (LNCS), pages 106–117, October 2003.