Internet and the Erlang formula

Thomas Bonald, James Roberts

To cite this version:

Thomas Bonald, James Roberts. Internet and the Erlang formula. Computer Communication Review, Association for Computing Machinery, 2012, 42 (1), pp.23-30. <hal-00941783>

Internet and the Erlang formula

Thomas Bonald[*]
Telecom ParisTech
Paris, France
thomas.bonald@telecom-paristech.fr

James Roberts
INRIA
Rocquencourt, France
james.roberts@inria.fr

ABSTRACT

We demonstrate that the Internet has a formula linking demand, capacity and performance that in many ways is the analogue of the Erlang loss formula of telephony. Surprisingly, this formula is none other than the Erlang delay formula. It provides an upper bound on the probability a flow of given peak rate suffers degradation when bandwidth sharing is max-min fair. Apart from the flow rate, the only relevant parameters are link capacity and overall demand. We explain why this result is valid under a very general and realistic traffic model and discuss its significance for network engineering.

Categories and Subject Descriptors

C.2.5 Local and wide-area networks –Internet, C.4 Performance of systems – modeling studies

General Terms

Design, Performance

Keywords

Erlang formula, Traffic, Congestion

1. INTRODUCTION

A recent report on the NSF program FIND (Future Internet design) concluded that an important open issue for future research is the identification of "Erlang formulas" for the Internet [8]. The Erlang formula, or Erlang *loss* formula, is used in engineering the telephone network. It gives the probability of call blocking on a trunk group as a function of the number of trunks and the offered traffic. It is the archetype of the relation between demand, capacity and performance whose understanding is essential for cost effective network engineering.

Currently, the Internet is engineered more by the use of pragmatic rules of thumb than by applying soundly based mathematical models like that which led to the Erlang formula. This leads not only to inefficiencies through inappropriate sizing but also to misconceptions about the effectiveness of traffic controls and their ability to support differentiated services. This is why we agree with the FIND report that identifying Internet Erlang formulas is indeed important. We believe, however, that much of the necessary research has already been performed and that the main problem is a lack of awareness of known results and their implications. Crucially, this research considers Internet traffic in terms of stochastic processes of packet, flow and session arrivals.

Our objective in this paper is to propose a candidate Erlang formula for the Internet. Surprisingly, the essential demand–capacity–performance relation turns out here to be none other than the so-called Erlang *delay* formula! This is clearly a surprising result since it is commonly believed that Internet traffic is so complex that it is practically impossible to characterize performance in a simple way. The Internet is used by a very large number of different applications and its traffic characteristics change continually as new applications gain popularity. Furthermore, while it is well-established that telephone calls arrive as a Poisson process, the arrival process of IP datagrams has been shown to exhibit much more complex, self-similar or fractal-like behavior that defies parsimonious modelling [15].

The basis of our claim is a model of Internet traffic where flows arrive on a network link according to a particular, realistic arrival process and dynamically share its bandwidth. Flows at the considered link have an intrinsic peak rate determined either by their end-systems or by bottlenecks elsewhere on their path. We assume bandwidth sharing is, at least approximately, max-min fair. The Erlang delay formula is then shown to upper bound the proportion of time a flow of given peak rate would suffer loss or have to reduce its rate below the peak.

This result is significant for several reasons. It provides a useful, relevant dimensioning criterion, ensuring for instance a streaming flow of given coding rate suffers negligible degradation. It shows performance is broadly robust with respect to detailed traffic characteristics other than link capacity and overall expected demand. It constitutes a solid basis on which to build effective network engineering practice and to elaborate additional performance results. The Erlang delay formula is thus potentially as important for dimensioning the Internet as the Erlang loss formula is important for dimensioning the telephone network.

It is not possible in this short paper to review the large body of related work on dimensioning the Internet. It is useful, however, to recall some work on Gaussian approximations, typified by the papers of Fraleigh *et al.* [9] and van den Berg *et al.* [2]. The former estimates packet delay properties while the latter adopts a dimensioning criterion based on the probability the incoming bit rate exceeds link capacity. Traffic in both is modeled as a Gaussian process, implicitly assuming therefore that the arrival rate of packets is independent of congestion. The present proposals, on the other hand, are meant to account for the "closed loop"

[*] This author has carried out the work presented in this paper at LINCS – www.lincs.fr.

nature of Internet traffic where the arrival process can be significantly modified by the action of congestion control.

We first recall the significance of the Erlang formula for the telephone network. Our general, flow-based Internet traffic model is then introduced and used to characterize so-called transparent, elastic and overload traffic regimes. Performance of a link subject to this traffic is then evaluated under the assumption that active flows share link bandwidth fairly. Lastly, the results of the evaluation are applied to derive the proposed "Internet Erlang formula".

2. ERLANG AND THE TELEPHONE NETWORK

We first recall the importance of the Erlang formula for the telephone network.

2.1 Historical note

A. K. Erlang was a mathematician working for the Copenhagen Telephone Company from 1909, some 30 years after the inauguration of the first public telephone network [7]. In parallel with some illustrious contemporaries in other countries, and in opposition to the opinion of many skeptics, he established that the analysis of telephone traffic was amenable to the mathematical modelling tools of probability theory. His celebrated formulas, as discussed below, relate performance to capacity and a simple measure of demand, enabling cost-effective network dimensioning and providing a rigorous basis for network design. It is remarkable that the formulas established for the technology and usage of a century ago are still applied by network operators today.

2.2 The Erlang loss formula

The Erlang loss formula, the B-formula, gives the probability of call blocking when N trunks are offered traffic A:

$$E_B(A,N) = \frac{\frac{A^N}{N!}}{1 + A + \cdots + \frac{A^N}{N!}}, \tag{1}$$

where A is the product, call arrival rate × mean call holding time. The formula is derived from a mathematical model that makes a number of assumptions about telephone switching and the nature of traffic, including the following:

1. call arrivals constitute a stationary Poisson process;
2. calls are blocked if and only if all trunks are busy;
3. blocked calls are cleared.

Some assumptions are *realistic*, being based on observable reality, others are merely *convenient*, enabling a simple formula when a more accurate model would be intractable. For instance, to assume Poisson arrivals is realistic over relatively short timescales (less than 1 hour, say) when calls are generated by a large user population. The second assumption is realistic for present day switches though in Erlang's day, all trunks could not usually be tested by every call. It is convenient to suppose blocked calls are cleared though in practice callers usually make repeat attempts. This means the B-formula is good for dimensioning (for low blocking) but not for analysing performance in overload.

It is well-known now that Erlang's B-formula is insensitive to the distribution of the call holding time. The blocking probability depends only on the simple average measure of offered traffic A. This insensitivity explains why the formula remains a precious dimensioning tool today despite significant changes in usage over the last century. It also underlies much of network engineering practice since it informs us that the essential measure of demand to be monitored and forecast is offered traffic A, even when the above assumptions may not be perfectly reasonable.

The Erlang formula reveals the important scale economies phenomenon of networking (see Figure 1): achievable load $A(1-E_B)/N$, for a given target blocking probability E_B, increases towards 100% with trunk group size N. This has two important consequences for large N, say $N \geq 100$:

1. dimensioning based on maximum load is adequate (e.g., simply applying a load limit of 80% ensures blocking is lower than 1%),
2. there is little scope for service differentiation (e.g., dimensioning to realize a 5% loss rate for low priority calls would bring little gain compared to a 1% rate for all).

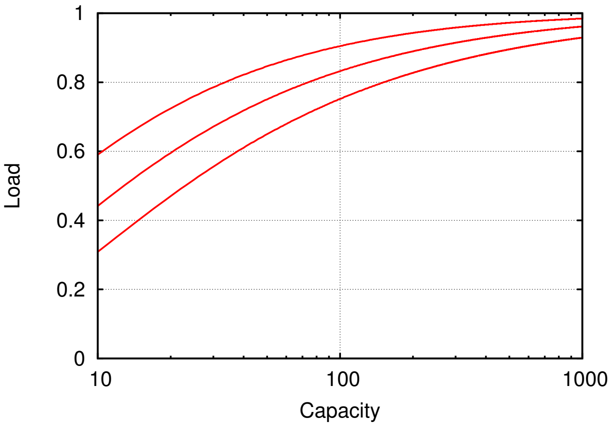


Figure 1: Achievable load as a function of trunk group capacity for blocking probabilities $5\%, 1\%, 0.1\%$ **(from top to bottom)**

Of course, the Erlang formula is not a universal tool that solves all telephone network engineering problems. It does, however, have a very important emblematic role through the lessons it provides about traffic modelling and the confidence it inspires that we are indeed able to relate demand, capacity and performance for a construction as complex as the telephone network.

2.3 Generalizations

The following two generalizations are actually more significant for the Internet than for the telephone network.

First, suppose calls do not occur as a Poisson process but are produced in "sessions". A session is a succession of calls separated by silent intervals (or think times). We assume sessions occur as a Poisson process. Under very general assumptions about the distributions of the number of calls per session, individual call holding times and silence intervals and their correlation, the call blocking probability is still given by Erlang B [3]. Under this Poisson session model the call arrival process would even be self-similar if the number of calls per session had a so-called heavy-tailed distribution.

The second generalization relates to heterogeneous call types distinguished by the number of trunks each call requires throughout its holding time. A call requiring c trunks is blocked and cleared if the number of free trunks on its arrival is less than c. Traffic for this type of call is defined by the product, call arrival rate × mean call holding time × c. Consider m types of calls, class-i calls requiring c_i trunks and offering traffic a_i. The probability that n trunks are occupied is proportional to $f(n)$, given by the following simple recurrence relation [11, 16]:

$$f(n) = \frac{1}{n}\sum_{i=1}^{m} a_i f(n - c_i), \tag{2}$$

for $n = 1, \ldots, N$, with $f(0) = 1$ and $f(n) = 0$ if $n < 0$. The blocking rate of class-i calls then follows as the probability that more than $N - c_i$ trunks are occupied. The formulas are valid under the same general assumptions as Erlang B, including the Poisson session model.

2.4 The Erlang delay formula

Before coming to the Internet, it is useful to recall a second result due to Erlang, the Erlang delay formula or C-formula, derived initially to dimension the number of operators manning a switchboard. The formula gives the probability $E_C(A, N)$ that a caller must wait when N operators receive offered traffic A, assuming $A < N$:

$$E_C(A, N) = \frac{\frac{A^N}{N!}\,\frac{N}{N-A}}{1 + A + \cdots + \frac{A^{N-1}}{(N-1)!} + \frac{A^N}{N!}\,\frac{N}{N-A}}. \tag{3}$$

This formula is valid under the following assumptions: call arrivals are Poisson, call service times have an exponential distribution, calls are served in arrival order and the queue length is unlimited. We show below that this formula has in fact much more general application in the Internet.

3. INTERNET TRAFFIC

Internet traffic is clearly much more complex than telephone traffic and the mix of applications that produces it continues to vary widely over time.

3.1 Packets, flows, sessions

Though the Internet protocols only deal with datagrams, it is important for network engineering to recognize that these belong to “flows” which in turn are components of “sessions”. For present purposes, a flow is defined as the succession of packets handled by a given link that relate to one instance of some application.

Flows are basically of two types:

- *elastic* flows download documents as fast as possible by adjusting their packet emission rate (e.g., through TCP) to use all available capacity,
- *streaming* flows, typically based on UDP, send packets as and when they are generated by the audio or video codec.

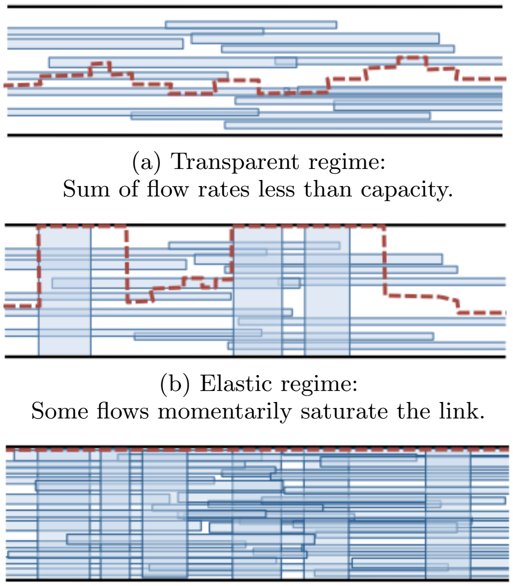
Each flow is characterized by some peak rate. For elastic flows, this peak rate is typically due to the access network, the server capacity or some other bottleneck on its path. For streaming flows, it is the peak rate of the codec. It matters little for the present discussion that some streaming flows may be emitted as progressive downloads in applications like YouTube.

A session is loosely defined as a set of flows that are related in some way. The session is in fact better defined by the requirement that any two sessions relate to independent activities, usually by distinct users. Sessions cannot in general be identified as such but, for the same reason a large population generates telephone calls as a Poisson process, it is natural that the arrival epochs of sessions using a considered network link are Poisson.

Poisson session arrivals have been observed experimentally by Paxson and Floyd for some kinds of session that can be identified in Internet trace data [15]. The same authors confirmed that flow and packet arrivals are anything but Poisson, on the other hand, and even exhibit the extreme correlation of self-similar processes. We explain later, in Section 5, that these characteristics are in fact only of secondary importance.

3.2 Traffic regimes

Figure 2 illustrates three traffic regimes that help to understand the scope for meeting performance requirements. Assuming flows have a constant peak rate, they can be represented simply in the figure as rectangles where height is peak rate and area is size. They share the bandwidth of a link represented by the parallel black lines. The dashed line represents the instantaneous overall input rate.



(a) Transparent regime:
Sum of flow rates less than capacity.

(b) Elastic regime:
Some flows momentarily saturate the link.

(c) Overload regime:
The link is permanently saturated.

Figure 2: Link occupancy regimes: rectangles represent flows (minimum duration × peak rate), the dashed line traces the sum of realized rates.

In the *transparent* regime, all flows have a relatively low peak rate and demand is such that, with very high probability, the sum of rates is less than link capacity. In this regime, packet loss is negligible and delays are tiny. Note that the Gaussian approximations proposed in [9] and [2] can be used for dimensioning if this regime prevails.

The *elastic* regime occurs when some flows have a peak rate that momentarily saturates the link. The buffer is then sure to overflow and flows suffer loss and delays that can be significant. Periods of transparency alternate with periods of saturation. Performance may be considered satisfactory if degradation can be confined to the high rate elastic flows.

The *overload* regime occurs when demand (flow arrival rate × mean flow size) exceeds link capacity. Performance is then typically very bad for all flows so that this regime needs to be avoided by appropriate traffic engineering.

3.3 Bandwidth sharing

Flows that are concurrently active on a given link may be said to share its bandwidth. In the transparent regime there is capacity to spare and every flow realizes its peak rate. A simple FIFO buffer is then sufficient to resolve contention between packets from distinct flows.

In the elastic regime some flows must reduce their rate. TCP normally realizes the necessary adjustment resulting in each flow receiving an allocation that is approximately max-min fair [13]. Strict max-min fairness would be realized if access to the link were controlled by a fair queuing scheduler [10]. It is worth recalling that fair queuing is provably feasible at any link rate, as long as demand overloads can be avoided [14]. With max-min fairness, only the high peak rate flows are constrained to reduce their rate. The others maintain their rate and suffer negligible loss, as if they experienced the transparent regime.

Bandwidth sharing in the Internet can be controlled to some extent by QoS mechanisms like Diffserv. It is possible, for example, to consider certain classes of traffic with priority, ensuring they experience a transparent or elastic regime even when other classes are in overload. It remains difficult to control performance, however, since this depends critically on the amount of traffic in each class and on the peak rates of its flows, parameters that are largely uncontrollable.

4. PERFORMANCE OF FAIR SHARING

We briefly present a number of recently obtained, powerful performance results for Internet traffic. We consider an isolated link in order to highlight the analogy with the Erlang loss formula. Note, however, that extensions equivalent to more general, known results for loss networks (e.g., see [12]) are discussed elsewhere [6].

4.1 Assumptions

As discussed above, max-min fair sharing between concurrent flows is a *realistic* assumption if routers impose per flow fairness [10]. It is just a *convenient* assumption if we must rely on end-system compliance in implementing congestion control.

We make the further *convenient* assumption that even streaming flows adjust their rate as necessary to respect fairness and that they preserve their volume (i.e., like elastic flows, if their rate is reduced below their peak rate they last longer). This simplifies modelling and the resulting approximation is accurate as long as the probability a streaming flow would suffer loss is small. As this should be an objective of dimensioning, this is similar in effect to Erlang's "blocked calls cleared" assumption.

We further adopt the *realistic* assumption that sessions occur as a Poisson process and the *convenient* assumption that on any given link, their flows occur singly in an alternating sequence with think times. Equivalently, we assume concurrent flows of the same session are considered as one for the purpose of bandwidth sharing. Flow sizes and think time durations are generally distributed and can be correlated. The number of flows in the same session has a general distribution.

4.2 A common peak rate

Consider a link of capacity C offered traffic A, in bit/s, and suppose each flow has the same peak rate c. Under the above traffic and sharing assumptions, the number of flows concurrently active x behaves like the number of customers in a multi-server processor sharing queue [1]. In particular, when C/c is an integer and under the stability condition $A < C$, the proportion of time a flow suffers congestion (in the sense that $xc \geq C$) is given by the Erlang C-formula, $E_C(A/c, C/c)$.

Thus, under the assumption of equal peak rates, the Internet Erlang formula is precisely the Erlang delay formula (3). Note that, due to the insensitivity property of the processor-sharing discipline, this formula is valid for the considered Poisson session traffic model with general flow size distribution while, for the FIFO queue envisaged by Erlang, it is only valid for Poisson flow arrivals and exponentially distributed call holding times.

4.3 A mixture of peak rates

In practice, flows in the Internet have a wide range of peak rates. Assume for convenience that the number of possible peak rates is limited to m, that these rates are $c_1, c_2, \ldots, c_m$ in increasing order and that flows of rate c_i offer traffic a_i. Let the number of active flows of class i be x_i and consider a time interval of congestion, where $\sum x_i c_i \geq C$.

With max-min fair sharing, the congestion is confined to flows of high rates. Specifically, there is an index j such that flows of classes j to m reduce their rate to a "fair rate" r satisfying $c_{j-1} < r \leq c_j$, while flows of classes 1 to $j-1$ maintain their peak rate. The precise value of r is such that the sum of realized flow rates is equal to C. It turns out that performance evaluation under max-min fairness is now analytically intractable.

To make progress it is useful to make a further *convenient* assumption. We assume that sharing is "balanced fair", as defined in [6, 5]. In the present context, this means that when $\sum x_i c_i \geq C$, all flows see a rate reduction, the reduction of flows of class i being approximately proportional to c_i. It has been proved that balanced fairness is the only policy for which it is possible to derive explicit performance results for general rate and demand vectors, $\{c_i\}$ and $\{a_i\}$, and that these results do not depend on any more detailed traffic characteristics.

Under balanced fairness, the probability that the total flow rate $\sum x_i c_i$ is equal to n, assuming C and the $\{c_i\}$ are integers, is proportional to a function $f(n)$ that satisfies the recurrence relations:

$$f(n) = \begin{cases} \frac{1}{n} \sum_{i=1}^{m} a_i f(n - c_i) & \text{if } n < C, \\ \frac{1}{C} \sum_{i=1}^{m} a_i f(n - c_i) & \text{if } n \geq C. \end{cases} \qquad (4)$$

Comparison of (2) and (4) reveals a quite remarkable parallel between loss systems on one hand and balanced fair systems on the other that in fact extends well beyond the results we are able to summarize here.

The function $f(n)$ can be used to derive a number of performance parameters like the congestion rate, the probability input rate $\sum x_i c_i$ exceeds capacity C. Others can be derived from further properties of balanced fairness discussed in [6]. Importantly, it has been verified by simulation that many performance results derived under the convenient balanced fairness assumption closely approximate those obtained for max-min as well as other fairness criteria. The balanced fairness assumption is then reasonable as well as convenient.

4.4 Throughput and congestion

It has been shown in particular that the expected throughput of a flow of peak rate c_i is approximately equal to the minimum of c_i and $C-A$ where $A = \sum a_i$ is overall demand. The fact that C is large and utilization A/C is typically not more than 80% explains why Internet backbone links rarely impact perceived performance. They are in the *transparent regime* since no flows are able to saturate the residual free capacity, i.e., $c_i << C - A$ for all i.

If the $\{c_i\}$ and $\{a_i\}$ are known, it is possible to use the recurrence relations (4) to dimension links to ensure a low congestion probability. This means the link stays with high probability in the transparent regime. This is only satisfactory however if the c_i are all guaranteed to be relatively small. Otherwise the variance of the offered traffic is high so that congestion can only be avoided by limiting mean load to a small fraction of capacity.

While the transparent regime prevails in the Internet, justifying the Gaussian traffic models of [9] and [2], we anticipate that the elastic regime will become more common as flow rates increase, notably between well-connected servers and data centers. In the next section we propose a simple alternative dimensioning criterion that makes no assumption about the c_i and uses only overall demand A. This is what we call the Internet Erlang formula.

5. THE INTERNET ERLANG FORMULA

We identify an explicit performance relation that, like the Erlang loss formula, involves only link capacity and expected demand. For all practical purposes, this relation is an upper bound on the probability of congestion and can be used to dimension Internet links.

5.1 Performance criteria

Following the above discussion, we consider the demand–capacity–performance relation with the following choice of performance criterion. We suppose a network provider seeks to limit the degradation suffered by streaming flows of peak rate no greater than c. Specifically, assuming max-min fair bandwidth sharing, the proportion of time P_c any currently active flow of rate c would suffer loss or have to reduce its rate should be less than some target ϵ. We refer to P_c as the rate-c congestion probability. The dimensioning objective, given demand A, is to provide sufficient capacity C such that $P_c < \epsilon$.

Under max-min fair bandwidth sharing, P_c is simply the probability that the fair rate is less than c. According to the model of the previous section, with some traffic mix defined by $\{c_i\}$ and $\{a_i\}$, we have:

$$P_c = \Pr\left(\sum_{i=1}^{m} x_i \min(c_i, c) \geq C\right). \tag{5}$$

5.2 A congestion probability bound

Unfortunately, max-min sharing is intractable: it is not possible to calculate the probability distribution of the fair rate. Moreover, any formula that depends on precise knowledge of $\{c_i\}$ and $\{a_i\}$ is hardly useful in practice since these data are not usually available. On the other hand, as shown below, there is a formula that, for all practical purposes, constitutes an upper bound on the congestion probability P_c and is valid for any traffic mix.

The test rate c divides the flows into two categories: "low-rate-flows" that have a peak rate less than c and "high-rate-flows" that have a peak rate greater than or equal to c. For given overall traffic A, P_c tends to increase either as the peak rate of low-rate-flows increases to c or as the peak rate of high-rate-flows decreases to c. In particular, the rate-c congestion for any traffic mix is upper bounded by that for traffic concentrated on rate c alone.

This statement is, in fact, not strictly true and precisely qualifying the sets of parameters and conditions for which it is remains an open research challenge. However, intuitive arguments, some mathematical demonstrations and the results of simulations lead us to the conviction to calculate P_c assuming uniform rate-c flows constitutes a valid, conservative approach for link dimensioning.

Reducing the rate of high-rate-flows while maintaining the same demand tends to increase the number of flows in progress. This naturally reduces the fair rate thus increasing P_c. Proposition 1 in the appendix proves this is generally true for any non-state dependent flow arrival process.

Consider now a set of classes such that $c_i \leq c$ for all i (i.e., after reducing the rate of high-rate-flows). The dimensioning objective is to ensure the link leaves the transparent regime with probability less than ϵ. We claim there is a folk theorem that such congestion increases with the "burstiness" of the arrival process. Assimilating burstiness to the variance of the input rate, this indeed increases with flow rates. It is, moreover, explicit in the Gaussian approach to dimensioning (e.g., see [2]) that congestion increases with input rate variance. Proposition 2 in the appendix proves the bound holds for *balanced fair* sharing and large C.

For all practical purposes, the worst case traffic mix for rate c congestion thus corresponds to all flows having the same peak rate, c. This is precisely the assumption of Section 4.2 where P_c was shown to be given by the Erlang C-formula. We conclude that a dimensioning rule to ensure $P_c < \epsilon$ for any traffic mix with overall demand A is to determine C such that $E_C(A/c, C/c) < \epsilon$. *The Internet Erlang formula is none other than the Erlang delay formula!*

5.3 Significance

Note that the Erlang C-formula provides a bound that is not necessarily tight. Figure 3 compares E_C with the results of simulations for a link of capacity 1000 shared by flows of peak rate 1, 10 and 100 arriving as a Poisson process[1].

[1] We simulate max-min fair sharing assuming Poisson arrivals and exponential flow size distributions. Statistics are

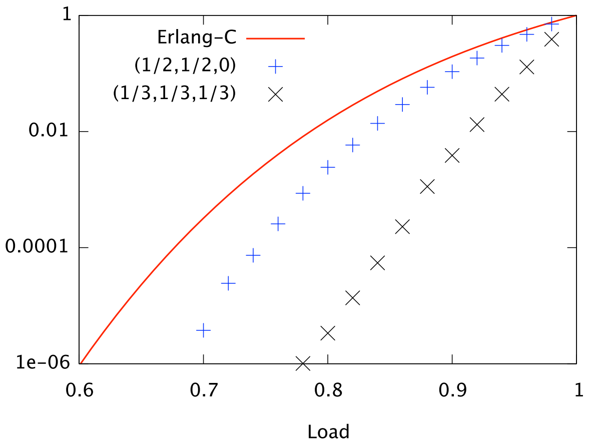


Figure 3: Congestion probability for rate-10 flows: Erlang C bound compared to simulations of a link of capacity 1000 with flows of rates 1, 10 and 100 contributing to load in proportions $(a_1/A, a_2/A, a_3/A)$ given in the legend.

The Erlang C-formula is used to bound the rate-c congestion probability for $c = 10$. The figure demonstrates that realized performance can be significantly better than that predicted by the bound, especially when there is a large proportion of high-rate-flow traffic.

The importance of the bound, even when it is not tight, is that it has precisely the same robustness as the Erlang loss formula. Performance depends only on overall expected demand A for whatever mix of flow rates and for the very general Poisson session traffic model. Recall that under this traffic model, packet and flow arrival processes are self-similar whenever the distributions of the flow size and the number of flows per session, respectively, have a heavy tail. The above results demonstrate that these characteristics have no significant impact on performance: the Erlang C bound is *insensitive*.

While the bound may not always be tight, it is nevertheless a very useful dimensioning tool. In particular, it exhibits scale economies similar to those of the telephone network, as depicted in Figure 4. Achievable utilization increases towards 100% as the ratio C/c grows. For example, utilization greater than 80% is compatible with 5 Mb/s streaming flows suffering congestion of less than 0.1% on any link of capacity greater than 1 Gb/s.

The Erlang C bound is a solid result that is independent of any assumption about traffic demand other than its overall average. If an operator knows more about the actual traffic mix, notably the traffic proportion due to high-rate-flows and their rates, a more precise demand–capacity–performance relation could be derived, using the mathematical models developed in [5] for instance.

6. CONCLUSIONS

We have demonstrated that, under a realistic flow-level model of Internet traffic, a simple performance parameter useful as a dimensioning criterion for network links depends only on link capacity C and overall demand A. Explicitly, the probability a flow of given peak rate c must reduce its rate is bounded by the Erlang delay formula, $E_C(A/c, C/c)$.

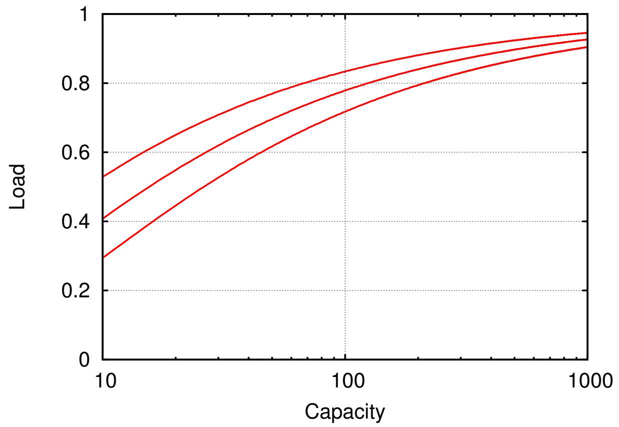


Figure 4: Achievable load A/C as a function of relative link capacity C/c for rate-c congestion probabilities $5\%, 1\%, 0.1\%$ (from top to bottom)

This Internet performance relation is analogous to the Erlang loss formula of the telephone network in several ways. It depends only on link capacity and expected load and not on more detailed traffic characteristics. It is therefore robust to changing usage. Both formulas reveal scale economies that validate simple, maximum-load dimensioning criteria for large capacity links. Its adoption as a dimensioning criterion facilitates network management since we only need to monitor and estimate average overall demand in representative busy periods.

Like the Erlang loss formula, however, the bound is not sufficient for all purposes and more precise performance measures are sometimes required. Our analysis is based on an extensive body of work, summarized in the paper [5], where many more results relevant to both wired and wireless networks are presented. It is, for example, possible to account for finite source traffic in the access network or to derive end-to-end performance measures for a network path.

The validity of the Internet Erlang formula relies on the assumption of max-min fair sharing. In practice, fairness does not need to be perfectly precise but one must question current reliance on end-systems voluntarily implementing TCP, or TCP-friendly, congestion control. We believe the future Internet should impose per-flow fairness. This is technically feasible and is arguably the only traffic control needed to satisfy performance requirements. Fair sharing makes the network manageable precisely because we then do have an Erlang formula. This cannot be said for other traffic control architectures, like Diffserv for instance.

7. REFERENCES

[1] S. Ben Fredj, T. Bonald, A. Proutière, G. Régnié and J. Roberts, Statistical bandwidth sharing: a study of congestion at flow level. In *SIGCOMM* 2001.

[2] H. van den Berg, M. Mandjes, R. van de Meent, A. Pras, F. Roijers, P. Venemans. QoS-aware bandwidth provisioning for IP network links. *Computer Networks*, 50, 631647, 2006.

derived for more than 10^8 arrival/departure events.

[3] T. Bonald, The Erlang model with non-Poisson call arrivals. In *SIGMETRICS / Performance* 2006.

[4] T. Bonald, J-P. Haddad, R. Mazumdar, Congestion in large balanced multirate links. In proceedings of *ITC 23* 2011.

[5] T. Bonald, L. Massoulié, A. Proutière and J. Virtamo, A queueing analysis of max-min fairness, proportional fairness and balanced fairness. Queueing Systems, 2006

[6] T. Bonald, A. Proutière, Insensitive bandwidth sharing in data networks. Queueing Systems, 2003.

[7] E. Brockmeyer, H. Halstrom and A. Jensen, Life and works of A.K. Erlang. *Transactions of the Danish Academy of Technical Sciences*, 1948.

[8] V. Cerf, B. Davie, A. Greenberg, S. Landau, D. Sincoskie, FIND Observer Panel Report. US National Science Foundation, 2009.

[9] C. Faleigh, F. Tobagi, C. Diot, Provisioning IP Backbone Networks to Support Latency Sensitive Traffic. In *Infocom* 2003.

[10] E. Hahne, Round-Robin Scheduling for Max-Min Fairness in Data Networks. *IEEE JSAC*, Vol 9, No 7, 1024–1039, 1991

[11] J. Kaufman, Blocking in a shared resource environment. *IEEE Trans. Commun.*, 29:500 1474–1481, 1981.

[12] F. Kelly, Loss networks. *Annals of Applied Probability*, 1, 319-378, 1991.

[13] F. Kelly, Mathematical modelling of the Internet. In "Mathematics Unlimited - 2001 and Beyond" (Editors B. Engquist and W. Schmid). Springer-Verlag, Berlin, 2001.

[14] A. Kortebi, L. Muscariello, S. Oueslati, J. Roberts, Evaluating the number of active flows in a scheduler realizing fair statistical bandwidth sharing. In *SIGMETRICS* 2005.

[15] V. Paxson and S. Floyd, Wide-area traffic: The failure of Poisson modeling in *SIGCOMM* 1994.

[16] J. Roberts, A service system with heterogeneous user requirement. *In* G. Pujolle, éd. : *Performance of Data Communications Systems and Their Applications*, p. 423–431, 1981.

APPENDIX

We show in two particular cases that the rate-c congestion probability P_c increases when high-rate-flow rates decrease to c or low-rate-flow rates increase to c while overall demand remains fixed.

Proposition 1. *Under max-min fair sharing, reducing the peak rate of high-rate-flows to c increases P_c for any $C > A$.*

Proof. We prove that reducing the rate of high-rate-flows to c can only increase the number of flows of each class. The result is then a direct consequence of the definition of P_c, (5).

The proof is based on the following coupling argument. Let x_i' be the number of class-i flows in the modified system where the peak rate of high-rate-flows is reduced to c. Assume that $x_i' \geq x_i$ for all i. We shall see that the rate of each flow is larger in the original system than in the modified system. If $\sum_i x_i' \min(c_i, c) < C$, then $\sum_i x_i \min(c_i, c) < C$ so that, in the original system, low-rate-flows realize their peak rate while high-rate flows have rates larger than c. Now if $\sum_i x_i' \min(c_i, c) \geq C$, let r' be the max-min fair rate in the modified system. We have:

$$\sum_{i=1}^{m} x_i \min(c_i, r') \leq \sum_{i=1}^{m} x_i' \min(c_i, r') = C,$$

so that the max-min fair rate in the original system cannot be less than r'. In both cases, the rate of each flow is larger in the original system than in the modified system. Now starting from any state such that $x_i' = x_i$ for all i and comparing both stochastic processes path-by-path, we conclude that $x_i' \geq x_i$ for all i. □

Proposition 2. *Assume $c_i \leq c$ for all i. Under balanced fair sharing, we have $P_c \leq E_C(A/c, C/c)$ for large enough C at any given load A/C.*

Proof. Since $c_i \geq c$ for all i, P_c is the congestion probability $\Pr(\sum_i x_i c_i \geq C)$. We use the asymptotic expression for the congestion probability derived in [4, Theorem 2]. Since we are interested in the limit for large C, we can assume that both the peak rates c_i and the target rate c are integers. Moreover, we let $c_1 = 1$ for some arbitrarily small demand a_1. We then have:

$$P_c \sim \frac{e^{-IC}}{\sqrt{2\pi C}\sigma} \sum_{i=1}^{m} \frac{a_i}{C - A} \frac{1 - e^{\tau c_i}}{1 - e^{\tau}}, \qquad (6)$$

where τ is the root of $\sum_i a_i e^{\tau c_i} = C$, $\sigma^2 = \sum_i a_i c_i e^{\tau c_i}$ and:

$$I = C\tau + \sum_{i=1}^{m} \frac{a_i}{c_i}(1 - e^{\tau c_i}).$$

It can be verified that for all $i > 1$:

$$\frac{\partial I}{\partial c_i} = -\frac{a_i}{c_i^2}(1 - e^{\tau c_i}(1 - \tau c_i)) < 0.$$

Letting a_1 tend to 0, we deduce that the minimal value of I is reached for $c_i = c$ for all i, which corresponds to the system with common peak rate c. The proof then follows from the fact that P_c is dominated by the exponential term e^{-IC}: we have $P_c \leq E_C(A/c, C/c)$ for large enough C. □