

This is the peer reviewed version of the following article: Tunnicliffe, Martin J. and Winnett, Maria (2009) An approximate stochastic analysis of the packet-pair probing technique for available bandwidth estimation. *International Journal of Communication Systems*, 22(6), pp. 651-669, which has been published in final form at <http://dx.doi.org/10.1002/dac.986>. This article may be used for non-commercial purposes in accordance with Wiley Terms and Conditions for Self-Archiving.

An Approximate Stochastic Analysis of the Packet-Pair Probing Technique for Available Bandwidth Estimation

Martin J. Tunnicliffe¹
Maria Winnett

*Faculty of Computing and Information Systems and Mathematics,
Kingston University, Kingston-on-Thames, Surrey KT1 2EE, U.K.
Tel.: +20 82417887 + 62674, E-mail: M.J.Tunnicliffe@king.ac.uk*

Abstract

The packet-pair probing algorithm for network-bandwidth estimation is examined and an approximate model is proposed for predicting its behaviour. The model replaces the Poisson arrival process with a Gaussian distribution and resolves the queue-size profile into two separate components: A transient component representing the buffer-emptying process and an equilibrium component representing the return to steady-state behaviour. Comparison with discrete-event simulation results shows that the model is accurate in single-hop paths when utilisation $\leq 70\%$ when the cross-traffic packets are $\leq \frac{1}{2}$ the size of the probe packets. When extended to two-hop paths, the model remains accurate for smaller cross-traffic packets ($\leq \frac{1}{10} - \frac{1}{5}$ the probe packet size).

1. Introduction

The term *available bandwidth* refers to the unused portion of a network path's capacity which new connections may utilise without taking bandwidth from the existing cross-traffic [1]. Reliable estimates of effective bandwidth are useful to network clients who may require a minimum bandwidth to support delay-sensitive multimedia applications, and to administrators for achieving optimal network configurations [2].

A network path typically consists of several links (or hops), each with its own capacity. The available bandwidth of the entire path is dictated by that of the *tight-link*, the link with the smallest available bandwidth. Available bandwidth a is related to tight-link capacity l , utilisation ρ and cross-traffic c by the formula

$$a = l - c = l \cdot (1 - \rho) \tag{1}$$

where $(1 - \rho)$ is the *idle-rate*, the ratio of time during which the link is inactive. Some bandwidth-probing algorithms determine a by multiplying l by the idle rate, which can be inferred from the delay distribution of probe-packets [3] or from information

¹ Corresponding author.

passed from the MAC layer [4]. Though accurate for links in isolation, these techniques are problematic in wider network settings [4].

An alternative approach is typified by the *packet-pair* technique [5]. This belongs to a family of algorithms in which short sequences of evenly spaced probe-packets are sent through the monitored path and increases in their temporal separation (or dispersion) are used to estimate the path's properties. The packet sequences (or pairs) are widely separated such the behaviour of each is unaffected by any of its predecessors and the disturbance to the network performance is minimal.

An advantage of the packet-pair technique is that it requires no prior knowledge of link capacity, which together with the effective bandwidth can (in principle) be inferred from the measured data. However, this rests upon certain assumptions concerning network behaviour; namely that packet scheduling is FIFO (first-in-first-out), that the raw bandwidth is constant and well-defined and that the bandwidth seen by discrete packet-pairs is identical to that enjoyed by sustained streams. None of these assumptions is universally valid in wireless and broadband access networks where methods based on the idle-rate produce more accurate results [3,4].

Nevertheless, the packet-pair algorithm is still valuable in conventional wired networks of switches and routers. Though it has been widely investigated, much of the previous work ignores the finite granularity of the cross-traffic which is treated as a continuous "fluid". While this is approximately valid under certain conditions, it can give rise to a "probing bias" [6] which distorts the bandwidth estimate. Attempts to analyse and eliminate this bias have tended to be quite complex (e.g. [7]) even when simplifying assumptions are made.

In the present paper we develop a simpler and more intuitive model, based partly on an empirical study of simulation data. The analysis mostly assumes a single-hop topology, though the extension to multi-hop topologies is also investigated. Simulations were performed using a purpose-written class-library (originally created for the work in [10]) which allows FIFO queuing nodes to be connected in arbitrary configurations and their status to be monitored using "virtual" (zero-size) packets. All the C++ classes are available online and can be downloaded from <http://staffnet.king.ac.uk/~ku12881/netclasses/>.

2. Packet-Pair Probing and the Fluid Model

2.1 The Fluid Approximation

The use of hypothetical fluids to approximate discrete data-flows has quite a long history. Essentially, individual packet arrivals are ignored in favour of their average arrival rate which is treated as a continuous variable. (Alternatively the discrete packets could be considered infinitesimally small, thus constituting a continuous "fluid".) This simplifies the mathematics, accelerating simulation speed [11] and bringing complex phenomena (such as TCP-flows [12]) within the scope of analysis. However, the disregard for individual packet behaviour sometimes introduces significant errors, as will shortly be demonstrated.

As mentioned before, available bandwidth is measured by injecting *probe-packets* into a path and observing their behaviour. The probe-packets' response can be analysed using three different models:

- Model 1:** The fluid approximation is applied to both the probe-traffic and the cross-traffic. (This was the approach originally used by Melander et al. [5].)
- Model 2:** The probe-traffic is considered discrete, while the cross-traffic is assumed to be fluidic.
- Model 3:** The probe-traffic and cross-traffic are both considered to be composed of discrete packets.

Models 1 and 2 are simple and are dealt with in the remainder of this section. Model 3 is more problematic and forms the main thrust of this paper.

2.2 Single-Hop Network Paths

During a probing event, packets are offered at a rate r bits/s and received at a measured rate m bits/s. Under the assumptions of Model 1, if the link capacity is l bits/s and the cross-traffic is c bits/s (Figure 1(a)), then the following relationship should hold:

$$m = \begin{cases} r; & r + c \leq l \\ l \cdot \frac{r}{r + c} & r + c > l \end{cases} \quad (2)$$

i.e. packets are carried frictionlessly unless the aggregate arrival-rate exceeds the link capacity, in which case the bandwidth is shared proportionally between the competing streams. (This assumes a policy of proportional fair queuing.) By combining Eqns.1 and 2 we obtain:

$$\frac{r}{m} = \begin{cases} 1; & r \leq a \\ \frac{1}{l} r + \left(1 - \frac{a}{l}\right); & r > a \end{cases} \quad (3)$$

Thus a can be detected as a “knee” in the graph of r/m vs. r , and l can be determined from the slope for $r > a$ (Figure 1(b)).

Model 2 requires that probe-packet arrivals be considered discrete events. The *packet-pair* technique employs sequences of just *two* probe-packets spaced Δ_{in} seconds apart, such that the offered rate $r = S_p / \Delta_{in}$ (where S_p is the probe packet size in bits). If the same two packets arrive Δ_{out} seconds apart at the network output, then the measured rate $m = S_p / \Delta_{out}$ and $r/m = \Delta_{out} / \Delta_{in}$. This is the *dispersion ratio*, the factor by which the packets' temporal separation is increased by the link.

Figure 2(a) shows the queue-size profile during the processing of a packet-pair: If the cross-traffic packets are much smaller than the probe packets (the fluid assumption) then they are served almost immediately on arrival and the equilibrium queue size is practically zero. If Probe Packet #1 arrives at time $t = 0$, the queue suddenly acquires S_p bits which are subsequently processed at a rate l bits/s and the packet completes service when $t = S_p/l$ seconds. Meanwhile, cross-traffic arrives at a rate c bits/s such that at time t seconds the buffer contains $n(t) = \max(S_p - (l - c)t, 0)$ bits. If Packet #2 arrives at $t = \Delta_{in}$ seconds, its service time must be $(n(\Delta_{in}) + S_p)/l$ and therefore

$$\frac{S_p}{l} + \Delta_{out} = \frac{n(\Delta_{in}) + S_p}{l} + \Delta_{in}. \quad (4)$$

Substituting $n(\Delta_{in}) = \max(S_p - (l - c)\Delta_{in}, 0)$ and re-arranging yields

$$\frac{\Delta_{out}}{\Delta_{in}} = \max\left(\frac{1}{l} \cdot \frac{S_p}{\Delta_{in}} + \frac{c}{l}, 1\right) \quad (5)$$

which (remembering that $r = S_p/\Delta_{in}$, $r/m = \Delta_{out}/\Delta_{in}$ and $c = l - a$) is identical to Eqn.3. Models 1 and 2 are therefore mathematically equivalent and will be collectively referred to as the “fluid approximation”.

Figure 3(a) shows the results of a simulated packet-pair probing of a single-hop path. (Probe packet size 500bytes and cross-traffic packet size $S_c = 1 \text{ byte}^2$ were used to justify the fluid approximation.) Least-square analysis of the upper portion of the graph yields link capacity and available bandwidth estimates accurate to within 1% of their true values.

2.3 Multiple-Hop Network Paths

While the results of Figure 3(a) were obtained using a single link in isolation, a real network-path may have two or more congestible links, each with its own available bandwidth. Consider the path A-B in Figure 4(a): Node 1 forwards data at l_1 bits/s and Node 2 at l_2 bits/s. Node 1 has the lowest link capacity ($l_1 = 1 \text{ Mbit/s}$) and is termed the *narrow-link* of the path. However, Node 2 carries $c_2 = 1.5 \text{ Mbit/s}$ cross-traffic so its available bandwidth a_2 is only 500kbit/s (compared with 800kbit/s at Node 1). Node 2 is therefore the *tight-link*, which dictates the overall effective bandwidth.

In a multi-hop path, each congestible link may generate its own characteristic slope-change in the Δ_{out}/Δ_{in} vs. r curve. Park et al. [7] suggested that the characteristics of any link downstream of the tight-link will be unobservable since the probe-packet separation is too wide to be further expanded. However, this is not necessarily true if

² The simulation software is not subject to the constraints of TCP/IP, whose packet size is typically limited to 46-1500bytes by the Ethernet protocol.

the tight-link capacity is greater than the available bandwidth of downstream links. Melander et al. [5] used this principle to analyse multi-link paths by iterative application of Eqn.2: Using the notation employed in Figure 4, the rate offered to Node 2 (when Node 1 is congested) is $l_1 r / (r + c_1)$ which is potentially capable of exceeding a_2 . If Node 1 (which lies closer to probe-source than Node 2) is the tight-link, the model becomes:

$$\frac{\Delta_{out}}{\Delta_{in}} = \begin{cases} 1; & r \leq a_1 \\ \frac{1}{l_1} r + \left(1 - \frac{a_1}{l_1}\right); & a_1 < r \leq a_2 \frac{l_1 - a_1}{l_1 - a_2} \\ \left(\frac{1}{l_1} + \frac{1}{l_2} - \frac{a_2}{l_1 l_2}\right) r + \left(1 - \frac{a_1}{l_1}\right) \left(1 - \frac{a_2}{l_2}\right); & r > a_2 \frac{l_1 - a_1}{l_1 - a_2} \end{cases} \quad (6)$$

The assumption that $a_2 > a_1$ (called “smallest surplus first” of SSF) [5] implies that the second slope-change occurs at a rate higher than a_2 , which is clearly the situation in Figure 4(b). However, if Node 2 is the tight-link then it congests *before* Node 1 and the model becomes:

$$\frac{\Delta_{out}}{\Delta_{in}} = \begin{cases} 1; & r \leq a_2 \\ \frac{1}{l_2} r + \left(1 - \frac{a_2}{l_2}\right); & a_2 < r \leq a_1 \\ \left(\frac{1}{l_1} + \frac{1}{l_2} - \frac{a_2}{l_1 l_2}\right) r + \left(1 - \frac{a_1}{l_1}\right) \left(1 - \frac{a_2}{l_2}\right); & r > a_1 \end{cases} \quad (7)$$

which is the situation illustrated in Figure 4(a). We shall call this the LSF or “largest surplus first” model.

Figure 3(b) compares the results of a simulated probing of the LSF (Figure 4(a)) network with the predictions of Eqn.7, showing close agreement within the three rate-domains for near-fluid traffic ($S_c = 0.002 S_p$). However, without any knowledge of the network configuration, the data could just as easily be interpreted in terms of the SSF model, in which case the topology shown in Figure 4(b) would be inferred. (This network, when simulated under near-fluidic conditions, produces results almost identical to those of Figure 4(a)).

Melander’s solution to this ambiguity was *always* to assume the SSF configuration [5]. This produces worst-case results, since the available bandwidths of the upper bottlenecks tend to be lower than those inferred from any alternative model. Fortunately the most important inference, namely the tight-link bandwidth, is the same under both models. Though the model can be extended to include any number of congestible links, the current paper will be limited to one and two-hop scenarios.

2.4 Limitations of the Fluid Model

The fluid approximation requires that cross-traffic packets be far smaller than the probe packets; if this is *not* the case, a greater-than-predicted dispersion is observed when $r \approx a$. This effect, sometimes called “probing bias” [6], is visible in the simulation results in Figure 5: Applying linear regression to such data tends to produce an overestimation of l and an underestimation of a .

To understand the origin the probing bias it is necessary to consider how Model 3 (the “true” scenario of discrete probe-traffic *and* discrete cross-traffic) differs from the fluid approximation. Firstly, discrete cross-traffic packets take a finite time to be serviced, so finite queues form even when $r < a$. This gives rise to an average “background” or equilibrium queue-size n_{eq} , which raises the zero reference-level of the queue-size profile (Figure 2(b)). Secondly, since traffic arrives in discrete randomly-timed packages, the queue-size profile $n(t)$ acquires a stochastic variability and it is necessary to talk in terms of the *mean* queue-size profile $\bar{n}(t)$ (which would be observed over many repeated probing events), together with a corresponding variance function. Thirdly, this variance creates a finite empty-queue probability once the probe-packet has been serviced, slowing the mean emptying rate as the queue approaches equilibrium and thus giving rise to the concave $\bar{n}(t)$ profile shown in the inset of Figure 5. This makes the service time of the second packet greater than it would have been under the linear (fluid) model, increasing the rate of dispersion.

Probabilistic models have provided more accurate predictions of the queuing response under discrete traffic: Park et al. [7] used the transient $M/D/c$ model developed by G.J. Franx [8] to predict the evolution of the state probability vector and hence the mean expected queue-size. This model assumes that cross-traffic arrival is governed by a stationary Poisson process, though the introduction of equivalent-rate Pareto ON-OFF traffic produced no major changes in system behaviour. Though highly accurate, the model is both complex and computationally intensive.

In the current paper we develop a simpler model of the queuing dynamics and apply it to the analysis of the packet-pair probing event. The model, which includes both intuitive reasoning and empirical observation, agrees quite closely with simulation data under moderate ($\leq 70\%$) utilisation when cross-traffic packets are relatively small ($S_c \leq \frac{1}{2}S_p$). We believe that with its relative simplicity, the model will be readily extendible to more complex probing situations.

3. Discrete Queuing Dynamics

3.1 Discrete Cross-Traffic Model

If a queue contains n_0 bits at time $t=0$ then at any time $t \geq 0$ there be $\max(n_0 - lt, 0)$ of the *original* n_0 bits remaining, while a number of new bits may have arrived. If we assume (along with Park et al. [7]) that the packet inter-arrival time is exponentially distributed and that all packets contain exactly S_c bits, then the number of arrived packets must be Poisson-distributed with a mean of ct/S_c and a

standard deviation of $\sqrt{ct/S_c}$. Therefore the number of arriving *bits* must have a mean value ct and a standard deviation $\sqrt{cS_c t}$. Before there is any significant probability of the queue becoming empty, the mean queue length is given by $n_0 - (l - c)t$ with standard deviation $\sqrt{cS_c t}$, and can be approximated by a Gaussian distribution:

$$f_i(n) = \frac{1}{\sqrt{2\pi cS_c t}} \exp\left[-\frac{(n - n_0 + (l - c)t)^2}{2cS_c t}\right]. \quad (8)$$

However, this only applies for $n \geq 0$ (since the queue cannot empty below zero) and the contribution to the mean occupancy from the transient queue-emptying phase is given by:

$$\begin{aligned} \bar{n}_{trans}(t) &= \int_0^{\infty} n f_i(n) dn = \sqrt{\frac{cS_c t}{2\pi}} \exp\left[-\frac{(n_0 - (l - c)t)^2}{2cS_c t}\right] \\ &+ \frac{n_0 - (l - c)t}{2} \left[1 + \operatorname{erf}\left(\frac{n_0 - (l - c)t}{\sqrt{2cS_c t}}\right)\right]. \end{aligned} \quad (9)$$

The $n < 0$ portion of the Gaussian distribution (Eqn.8) represents the set of possibilities in which the queue has already completely emptied and is recovering its equilibrium behaviour (see Figure 6). The latter may be modelled as a standard *M/G/1* queuing system, for which the first and second moments of the waiting time w are given by:

$$E(w) = \frac{\lambda E(t_s^2)}{2(1 - \rho)} \quad \text{and} \quad E(w^2) = \frac{\lambda E(t_s^2)}{(1 - \rho)} E(w) + \frac{\lambda E(t_s^3)}{3(1 - \rho)}$$

where $\lambda = c/S_c$ is the arrival rate of packets per second, t_s is the packet service time and $\rho = c/l$ is the utilisation [9]. Since for an *M/D/1* system the packet service times are all equal ($t_s = S_c/l$), the mean equilibrium queue-size is

$$n_{eq} = l \cdot E(w) = \frac{cS_c}{2(l - c)} \quad (10)$$

and the standard deviation

$$\sigma_{eq} = \sqrt{l^2 E(w^2) - n_{eq}^2} = n_{eq} \sqrt{1 + \frac{4}{3} \left[\frac{l}{c} - 1\right]}. \quad (11)$$

Since equilibrium is not achieved at the instant the queue becomes empty we make a further assumption; namely that the equilibrium queue-size is restored abruptly t_{eq} seconds (the *effective equilibrium time*) after n reaches zero. (In other words, $n(\tau) = n_{eq} \cdot H(\tau - t_{eq})$ where τ is the time since the queue became empty and $H(t)$ is

the unit step function.) Thus the contribution to mean occupancy from equilibrium recovery for $t > t_{eq}$ is given by:

$$\bar{n}_{eq}(t) = n_{eq} \int_{-\infty}^0 f_{t-t_{eq}}(n) dn = \frac{n_{eq}}{2} \left[1 - \operatorname{erf} \left(\frac{n_0 - (l-c)(t-t_{eq})}{\sqrt{2cS_c(t-t_{eq})}} \right) \right] \quad (12)$$

and the overall mean queue-occupancy becomes:

$$\bar{n}(t) = \begin{cases} \bar{n}_{trans}(t); & t \leq t_{eq} \\ \bar{n}_{trans}(t) + \bar{n}_{eq}(t); & t > t_{eq} \end{cases} \quad (13)$$

Of course, the transition between the empty and equilibrium conditions is in reality gradual; replacing $\bar{n}(t)$ with an abrupt step function (see inset in Figure 7) is equivalent to replacing its first derivative $\bar{n}'(t)$ with a delta-function positioned at its centroid. Thus we define the effective equilibrium time as follows:

$$t_{eq} = \frac{1}{n_{eq}} \int_0^{\infty} t \cdot \bar{n}'(t) dt = \lim_{T \rightarrow \infty} \left[T - \frac{1}{n_{eq}} \int_0^T \bar{n}(t) dt \right]. \quad (14)$$

Profiles for $\bar{n}(t)$ were obtained for various combinations of l and S_c by averaging the results of 1,000 independent simulations. To determine t_{eq} , the integral in Eqn. 14 was computed between zero and the earliest instant at which $\bar{n}(t)$ exceeded n_{eq} . Figure 7 shows the results normalised in terms of the corresponding numbers of packet service-times $g = t_{eq}/t_s$ which is a function of utilisation only. Thus

$$t_{eq} = \frac{S_c}{l} \cdot g(\rho) \quad \text{where} \quad g(\rho) = \frac{0.333}{(1-\rho)^{2.19}} \quad (15)$$

where the expression for $g(\rho)$ was obtained empirically from the data.

3.2 Response to a Probe Packet Arrival

Suppose that the first probing packet (arriving at time $t = 0$) encounters the queue in its equilibrium condition, such that n_0 has a mean value $(S_p + n_{eq})$ and a variance σ_{eq}^2 . Eqns.9 and 12 can now be rewritten:

$$\begin{aligned} \bar{n}_{trans}(t) &= \sqrt{\frac{cS_c t + \sigma_{eq}^2}{2\pi}} \exp \left[-\frac{(S_p + n_{eq} - (l-c)t)^2}{2(cS_c t + \sigma_{eq}^2)} \right] \\ &+ \frac{S_p + n_{eq} - (l-c)t}{2} \left[1 + \operatorname{erf} \left(\frac{S_p + n_{eq} - (l-c)t}{\sqrt{2(cS_c t + \sigma_{eq}^2)}} \right) \right], \end{aligned} \quad (16)$$

$$\bar{n}_{eq}(t) = \frac{n_{eq}}{2} \left[1 - \operatorname{erf} \left(\frac{S_p + n_{eq} - (l-c)(t-t_{eq})}{\sqrt{2(cS_c(t-t_{eq}) + \sigma_{eq}^2)}} \right) \right] \quad (17)$$

which may be inserted into Eqn.13 to obtain the overall mean profile $\bar{n}(t)$. We have assumed of course that the convolution of the Poisson arrival distribution and the equilibrium occupancy distribution is approximately Gaussian and that the stepwise model for equilibrium recovery is valid under all loading conditions: The validity of these assumptions was tested *a posteriori* by comparing the model's predictions with simulation data: Figure 8 shows that the model loses accuracy under very heavy utilisation, and that the error increases as S_c approaches the magnitude of S_p . The fact that $\bar{n}(t)$ is under-predicted in regions where according to the model equilibrium recovery should not yet have begun (the onset of equilibrium-recovery is visible as a slight "glitch" in the curve) suggests that the abrupt-step approximation is partly responsible for the error. However, for all $S_c \leq \frac{1}{2}S_p$ and $\rho \leq 0.7$ the model agrees very closely with the simulation data.

4. Discrete Packet-Pair Model: Single Hop Scenario

We return now to the discrete-traffic packet-pair mechanism illustrated in Figure 2(b): Packets #1 and #2 arrive Δ_{in} seconds apart, and the time-separation between their departures Δ_{out} is measured. It is clear from the figure that:

$$\frac{n_{eq} + S_p}{l} + \Delta_{out} = \Delta_{in} + \frac{\bar{n}(\Delta_{in}) + S_p}{l} \quad (18)$$

which yields the following expression for the packet dispersion as a function of the probing rate $r = S_p / \Delta_{in}$:

$$\frac{\Delta_{out}}{\Delta_{in}} = 1 + \frac{r}{S_p l} \left\{ \bar{n} \left(\frac{S_p}{r} \right) - n_{eq} \right\} \quad (19)$$

where the function \bar{n} is obtained by substituting Eqns.16 and 17 into Eqn.13. Figure 9 compares simulation data obtained using 500byte probe packets with 250 and 100byte cross-traffic with the predictions of Eqn.19 and the fluid approximation. For the 100byte data, the discrete model is near-perfect, and converges with the fluid model for both large and small r . For the 250byte data the discrete model is mostly accurate but fails to converge with the fluid model under large r . The reasons for this will be addressed below:

4.1 The Limit as $S_c/S_p \rightarrow 0$

As S_c becomes infinitesimal relative to S_p , $n_{eq}, \sigma_{eq}, t_{eq} \rightarrow 0$ (Eqns.10 11 and 15) and Eqn.13 becomes:

$$n(t) = \begin{cases} S_p - (l-c)t; & t < S_p/(l-c) \\ 0; & t \geq S_p/(l-c) \end{cases} \quad (20)$$

(since $\lim_{x \rightarrow \pm\infty} [erf(x)] = \pm 1$). Under these conditions it is easy to show that

$$\frac{\Delta_{out}}{\Delta_{in}} = \begin{cases} 1 + \frac{r}{S_p l} \left\{ S_p - (l-c) \frac{S_p}{r} \right\} = \frac{r}{l} + \frac{c}{l}; & r > l-c \\ 1 & r \leq l-c \end{cases} \quad (21)$$

which is identical to Eqns. 3 and 5, so the discrete and fluid models converge (as expected) under near-fluid conditions.

4.2 The Limit as $r \rightarrow 0$

This corresponds to the limit as $t \rightarrow \infty$ in Eqn.13, which requires $\bar{n}(t) \rightarrow \bar{n}_{eq}(t) \rightarrow n_{eq}$ (Eqn.17) and hence $\Delta_{out}/\Delta_{in} \rightarrow 1$ (Eqn.19). This is consistent with Figure 9 which shows the discrete and fluid models converging under low probing rates.

4.3 The Limit as $r \rightarrow \infty$

This corresponds to the limit as $t \rightarrow 0$ in Eqn.13, which requires that $\bar{n}(t)$ be identical to $\bar{n}_{trans}(t)$ (Eqn.16). For small S_c ($\leq \frac{1}{2} S_p$) this is almost undistinguishable from the fluidic queue-size profile $S_p + n_{eq} - (l-c)t$, and hence the discrete and fluid models converge (Figure 9(b)). However, Figure 9(a) suggests that the discrete and fluid models do *not* converge exactly for larger S_c ($\approx \frac{1}{2} S_p$), under which conditions the queue-size profile predicted by Eqn.16 has a steeper slope than that of the fluid approximation causing an upward shift in the asymptotic dispersion rate. However, this weakness in the model is only significant under very high utilisation, and when the probe packets are not significantly larger than those of the cross-traffic.

5. Discrete Packet-Pair Model: Multiple-Hop Scenarios

To extend the analysis to multiple-hop scenarios we follow Melander et al. [5] in applying the one-hop model iteratively. Figure 10 illustrates this idea for a two-hop path: The input probe gap Δ_{in} is substituted into the model for Hop 1, the output of which (Δ_{out}^1) becomes the input gap (Δ_{in}^2) of Hop 2. This is perfectly valid under the fluid-traffic assumption (Section 2.2) where all the links behave deterministically. However, this is not the case with our discrete-traffic model, which takes a determinate Δ_{in} and calculates the *mean* value of the resulting output-gap *distribution*. To make $\Delta_{out}^1 = \Delta_{in}^2$ is to treat the latter as a determinate quantity, which is not strictly valid since nonlinearity within the Hop 2 may cause the resulting Δ_{out}^2 to differ from the true mean value. (To understand this, consider that $f(E(x)) \neq E(f(x))$ when $\text{var}(x) > 0$ and $f(x)$ is a nonlinear function.) We

nonetheless proceed with the approach, knowing that it is not altogether rigorously correct.

Figure 11 compares the model's predictions with simulations of the network path of Figure 4(a) and its "alias" topology of Figure 4(b), using 500byte probe packets and a variety of cross-traffic packet sizes. Despite its imprecise assumptions, the model agrees quite closely with the data for $S_c=50$ and 100bytes, even when the effects of the two slope-changes overlap. However, the introduction of 200 and 250byte packets yields more significant errors, which can be explained by the greater statistical variation created by the larger packets.

It is interesting to consider a third scenario where the two links have equal available bandwidth, and there is no single identifiable tight-link. Figure 12 shows an example: Although Node 1 is the narrow-link, both nodes have exactly 700kbit/s of available bandwidth. Again 500byte probe packets were used in conjunction with 100 and 250byte cross-traffic packets: The model agrees closely with the 100byte simulation results, while the 250byte simulation yields a significantly greater dispersion than the model.

6. General Discussion and Conclusions

This paper began by considering the probing of single and multi-hop network paths under the packet-pair algorithm. Having demonstrated the breakdown of this model under finite packet-sized cross traffic, an approximate stochastic model was developed and tested. This model was shown to produce accurate results for single-hop networks, though errors appeared when the utilisation was high (80-90%). Furthermore, these errors became more severe as the cross-traffic packet size increased: For 80% utilisation the model was reasonably accurate when $S_c = \frac{1}{5} S_p$ but significantly inaccurate when $S_c = \frac{1}{2} S_p$. When applied to two-hop network paths, the model proved accurate only for smaller S_c (typically $\frac{1}{10}$ to $\frac{1}{5} S_p$). Such errors are to be expected, given the underlying assumptions: Firstly the "true" Poisson and approximate Gaussian distributions are only similar when the standard deviation is small relative to the mean (which ceases to be true when S_c approaches S_p .) Secondly, high utilisation causes the queue-size profile to decay over longer time-periods, over which a better representation of equilibrium recovery than the simple step-model (Figure 7) is required. Finally the iterative application of the model to two-hop paths requires that statistical variation in the output-gap of the first hop be small. Increases in S_c tend to increase this variability, thus reducing the accuracy of the prediction. However, if S_p is kept relatively large (say 1500bytes in an Ethernet network) the model should be usable for typical average packet sizes and practical levels of utilisation ($\leq 60\%$).

The paper has mostly addressed the modelling of the probing experiment rather than the extraction of network-path information from packet-pair data. The latter is more challenging, and will be addressed in a separate paper. One particular challenge concerns the large stochastic variance of individual packet dispersions: Since many readings (100+) are often needed to obtain a reliable average, measurements must be spread over large time-windows and short time-scale behaviour (particularly

variations in the cross traffic c) cannot be observed. The measurement window can of course be reduced by bunching the packet-pairs closer together, but this introduces problems of its own; namely the interference between probing events compromises their statistical independence and the probe traffic takes an unacceptable share of the bandwidth being measured.

The software used to test the model was based on simple queuing assumptions which may not perfectly represent many real network components: In particular, a single raw bandwidth and FIFO-queuing at each node, assumptions which are not universally valid in access networks and wi-fi [4]. The model (and its successors) will need to be verified using a more realistic network simulator and/or hardware components.

References

1. Dovrolis,C Ramanathan,P, Moore,D, What do Packet Dispersion Techniques Measure?, *Proc. IEEE INFOCOM 2001*, 2, Anchorage, AK, USA, April 2001, pp.905-14.
2. Crovella,M Krishnamurthy,M, *Internet Measurement: Infrastructure, Traffic and Applications*, Wiley, 2006, p.127-36.
3. Lakshminarayanan,K, Padmanabhan,V.N., Padhye,J, Bandwidth Estimation in Broadband Access Networks, *Proc. 4th ACM SIGCOMM Internet Measurement Conference (IMC'04)*, Taormina, Italy, October 2004, pp.314-21.
4. Lee,H.K., Hall,V, Yum,K.H., Kim,K.I., Kim,E.J., Bandwidth Estimation in Wireless Lans for Multimedia Streaming Purposes, preprint available at <http://www.cs.utsa.edu/~yum/publication/icme06.pdf>, 2007. (Accessed 17 September 2008.)
5. Melander,B, Björkman,M, Gunningberg,P, A New End-to-End Probing and Analysis Method for Estimating Bandwidth Bottlenecks, *Proc. IEEE Globecom'00*, 1, San Francisco, CA, USA, November 2000, pp.415-20.
6. Liu,X, Ravindran,K, Liu,B, Loguinov,D, Single-Hop Probing Asymptotics in Available Bandwidth Estimation: Sample-Path Analysis, *Proc. 4th ACM Internet Measurement Conference*, Taormina, Italy, October 2004, pp.300-13.
7. Park,K-J, Lim,H, Choi,C-H, Stochastic Analysis of Packet-Pair Probing for Network Bandwidth Estimation, *Computer Networks*, 50, 2006, pp. 1901-15.
8. Franx,G.J., The transient M/D/c queueing system, preprint available at <http://www.math.vu.nl/sto/publications/2002-9.pdf>, 2002. (Accessed 17 September 2008.)
9. Scholl,M, Kleinrock,L, On the M/G/1 Queue with Rest Periods and Certain Service-Independent Queuing Disciplines, *Operations Research*, 31, 1983, pp.705-19.
10. Hosseinpour,M, Tunnicliffe,M.J. TOPP Probing of Network Links with Large Independent Latencies, *Proc. 8th. Annual Postgraduate Symposium on Telecommunications, Networking and Broadcasting*, Liverpool, UK, June 2007, pp.381-6.
11. Tunnicliffe,M.J., Parish,D.J., A Hybrid Simulator for an ATM Network, *Int. J. of Commun. Syst.*, 13, 2000, pp.179-84.
12. Liu,J, Packet-Level Integration of Fluid TCP Models in Real-Time Network Simulation, *Proc. 38th. Conference on Winter Simulation*, Monterey, CA, USA, December 2006, pp.2162-9.

TOP OF FIGURE

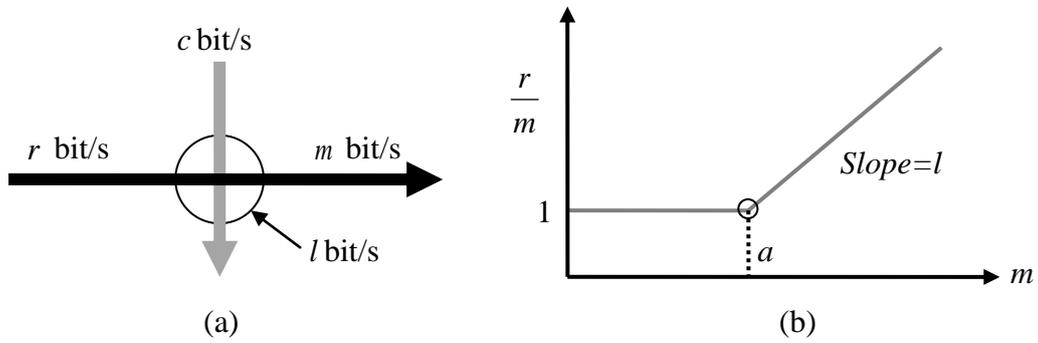


Figure 1: Interaction of fluidic traffic in a link of finite capacity l bits/s. (a) The offered probe rate r bits/s is reduced to a measured rate m bits/s by its passage through the link. (b) The “dispersion ratio” r/m is equal to 1 when there is no congestion ($r < a = l - c$) but increases linearly when the available bandwidth is exceeded.

TOP OF FIGURE

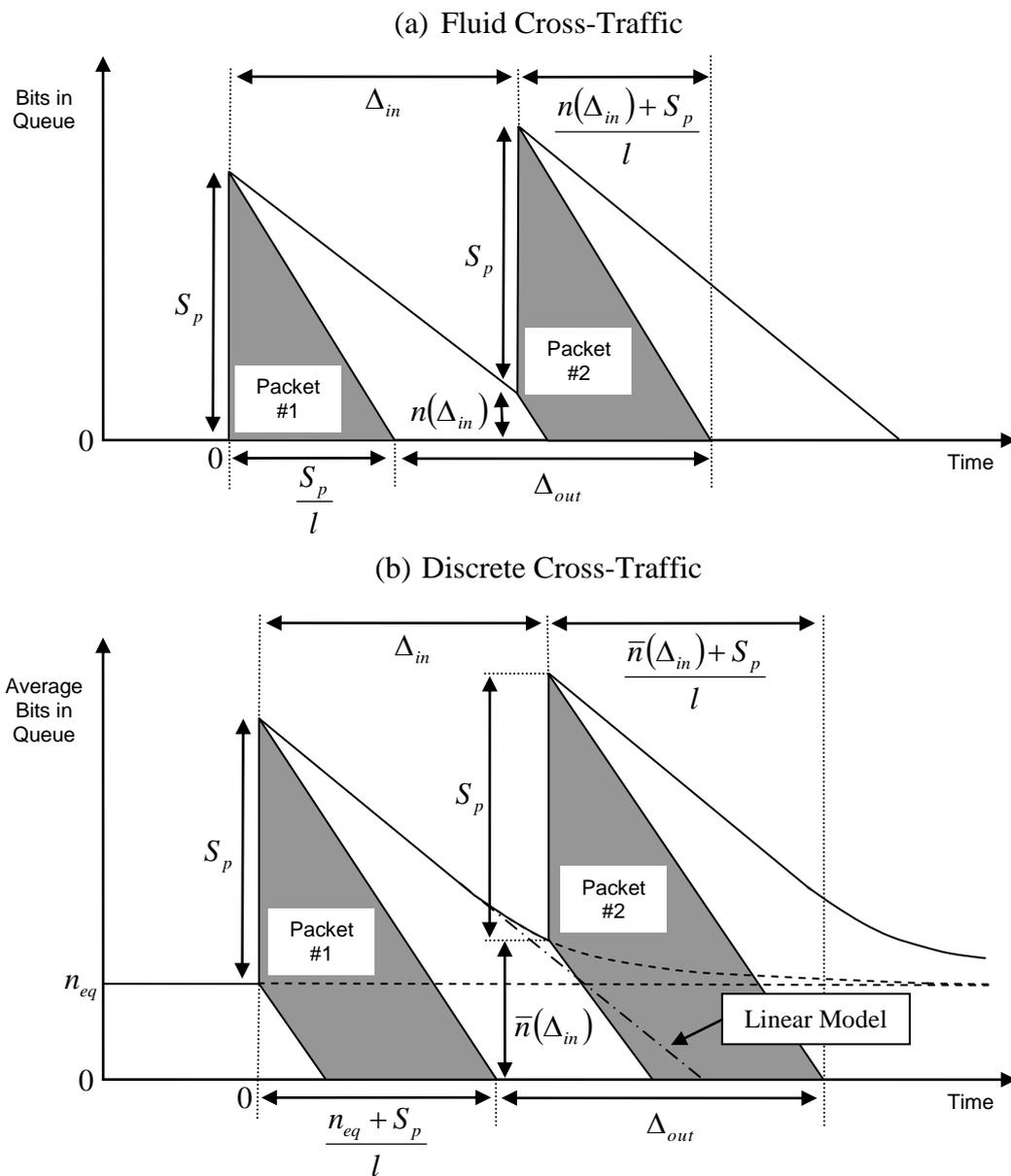
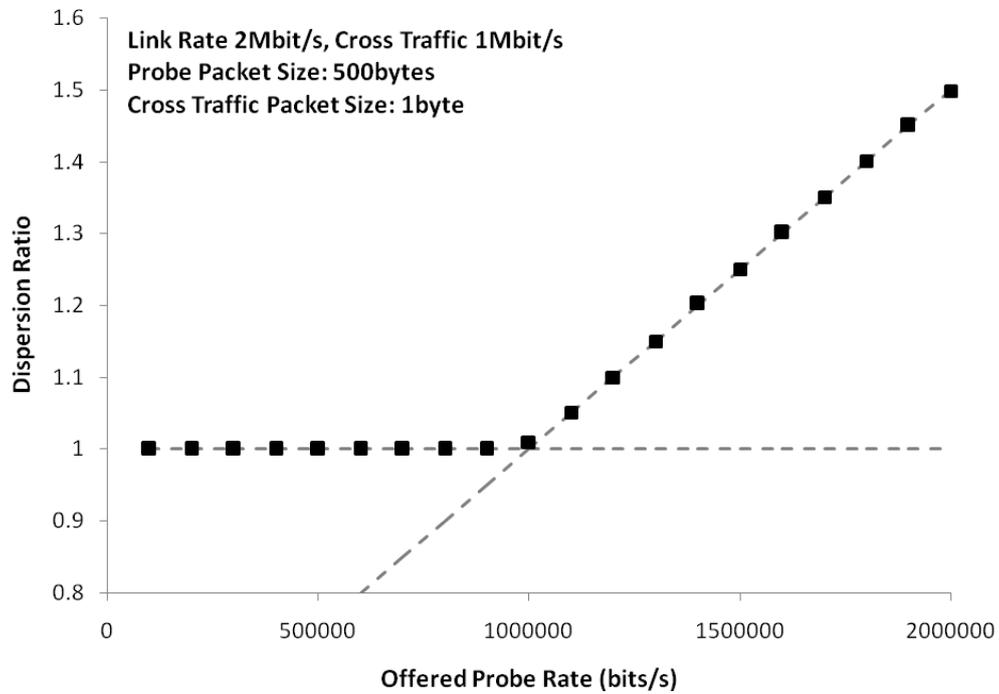


Figure 2: Queue-size profiles during a packet-pair probing event. Packet #1 creates a disturbance in the mean queue size which decays over time. The remnant is sensed in terms of the additional time taken to service Packet #2.

TOP OF FIGURE

(a) Single-Node Network Path



(b) Two-Node Network Path

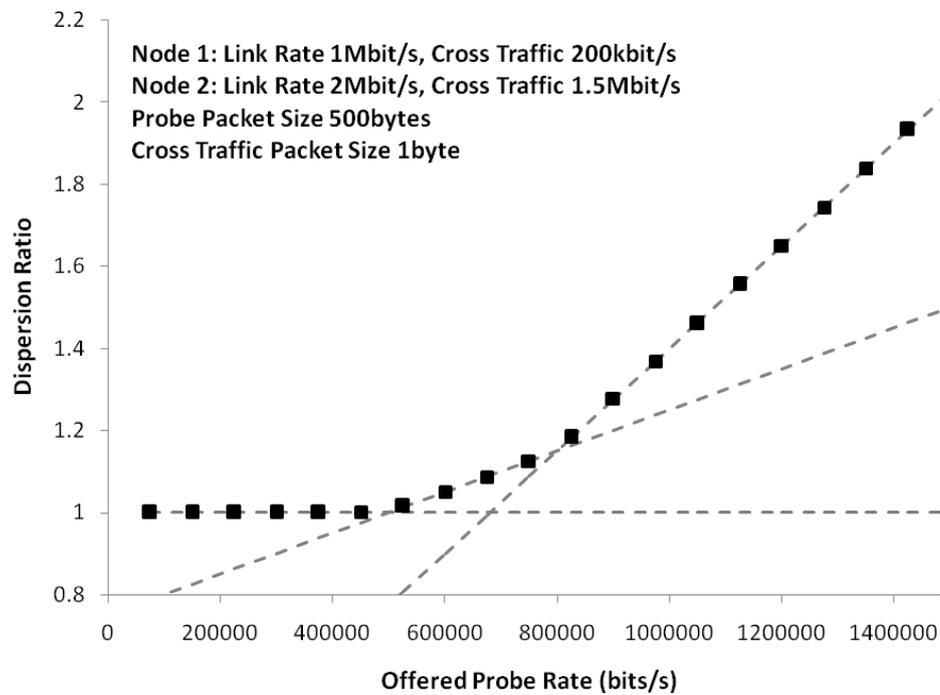


Figure 3: Simulated dispersion-rate profiles obtained by probing (a) a single-hop and (b) a two-hop path under near-fluidic conditions (500byte probe packets with 1 byte cross-traffic). The broken lines indicate the fluid predictions for the different domains of Eqns. (3) and (7).

TOP OF FIGURE

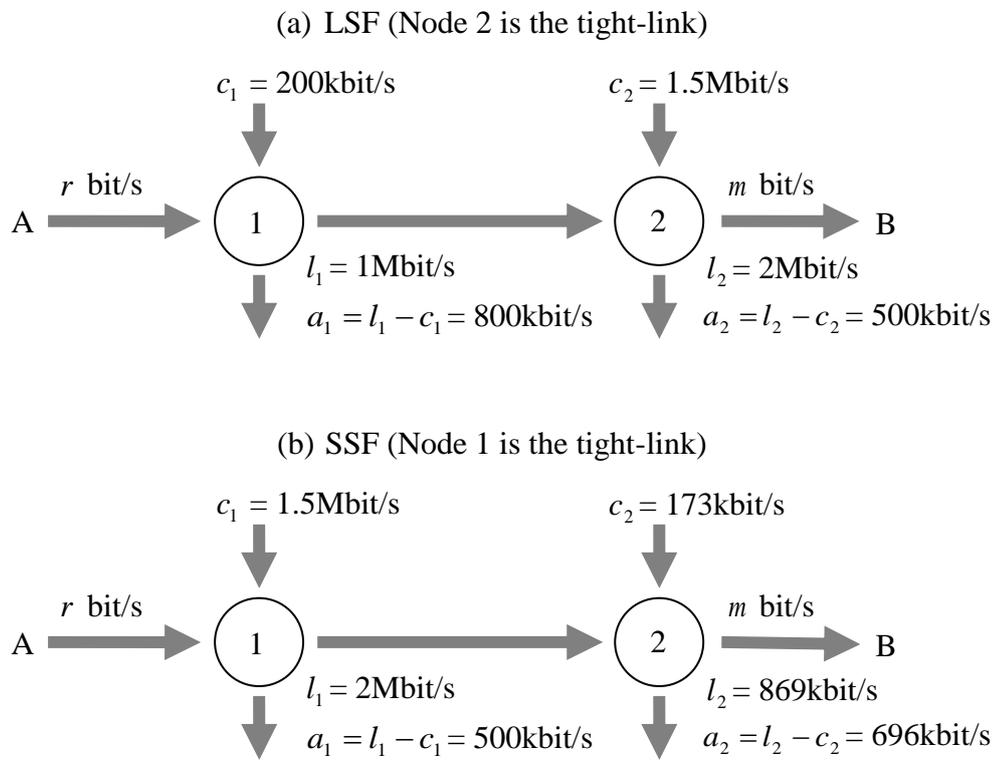


Figure 4: Two-hop network paths: (a) Largest Surplus First (LSF) and (b) Shortest Surplus First (SSF).

TOP OF FIGURE

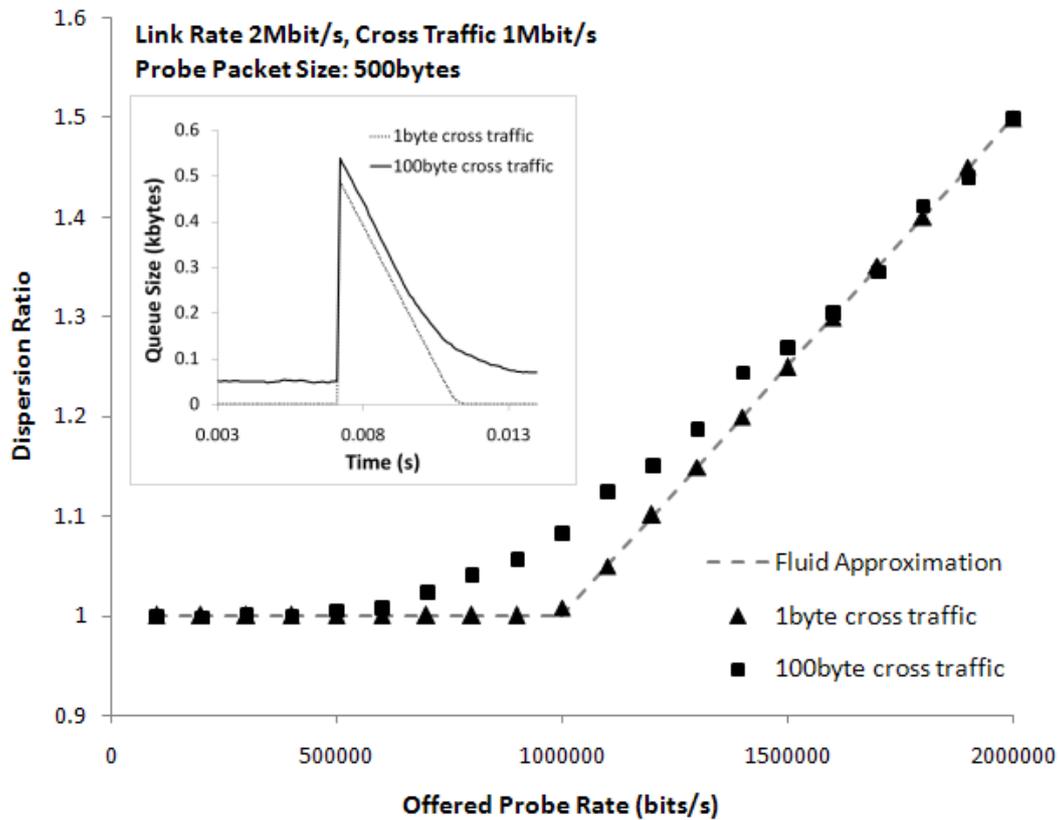


Figure 5: Simulated dispersion rate profiles for a single-hop path carrying 1 and 100byte cross-traffic packets, compared with the corresponding fluid approximation. (Each data point represents the average of 500 measurements.) While the 1byte results are almost identical to the fluid model, the 100byte results show a consistent upward bias when the offered rate is close to the available bandwidth (1Mbit/s). The inset shows the corresponding average queue-size profiles during the passage of a probe packet. (Each graph represents the average of 1000 simulations.)

TOP OF FIGURE

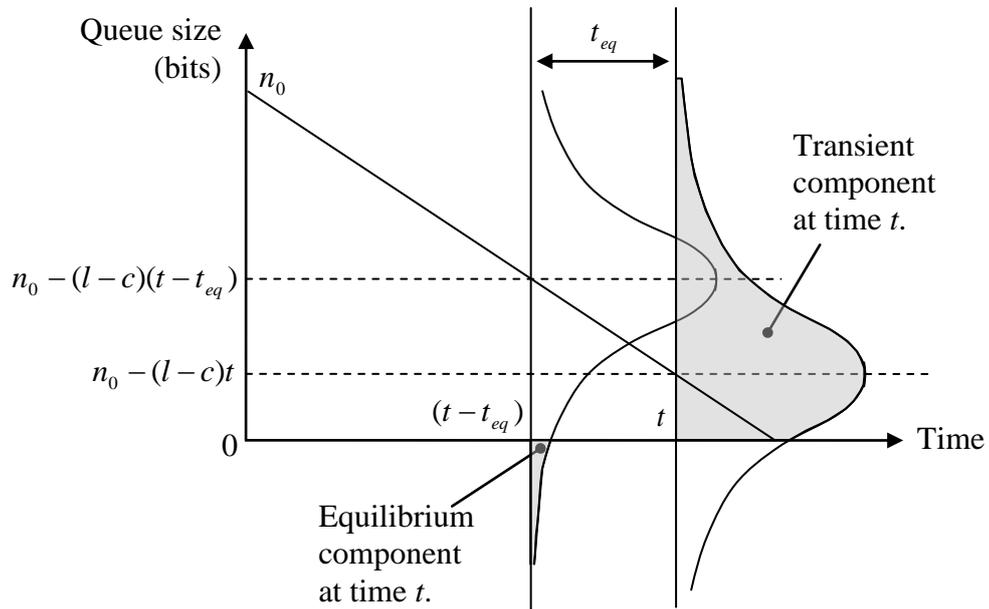


Figure 6: Schematic representation of the Gaussian queue-size pdf's during queue-emptying, showing the transient and equilibrium components. The equilibrium time t_{eq} represents the effective time-lag between the queue becoming empty and returning to its equilibrium mean occupancy.

TOP OF FIGURE

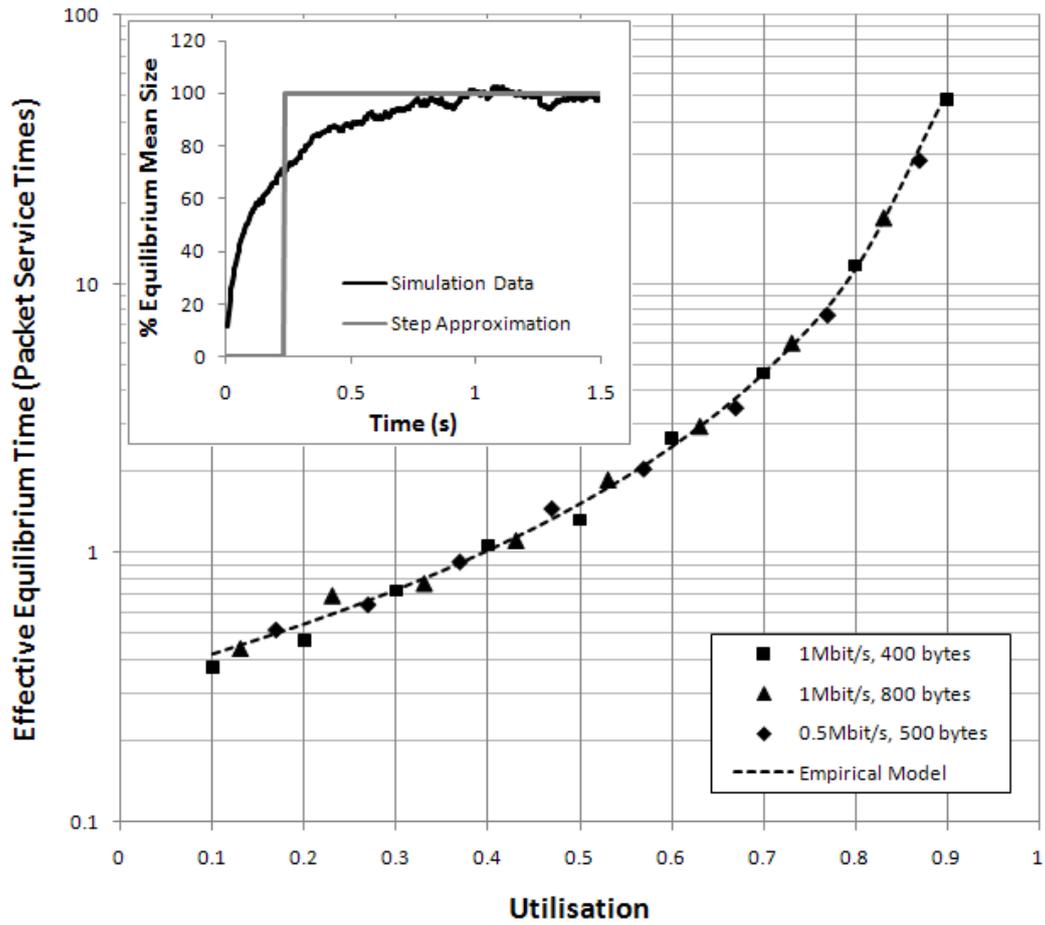
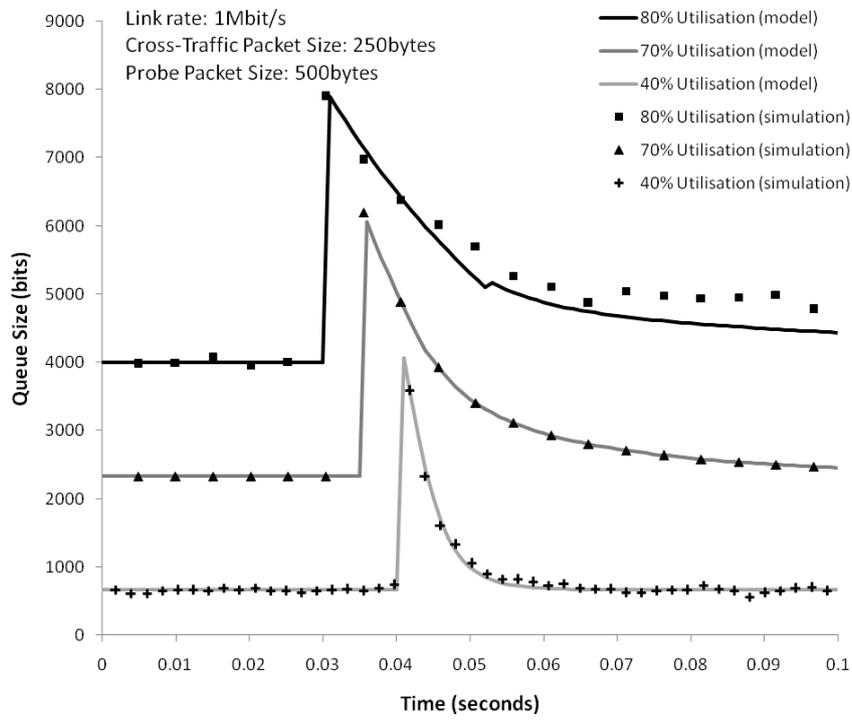


Figure 7: Effective equilibrium time (transition time for the step approximation, see inset) as a function of utilisation. Results were obtained using three combinations of packet size and server rate. Each data point represents the mean of five runs of 1,000 independent simulations.

TOP OF FIGURE

(a) $S_c = \frac{1}{2} S_p$



(b) $S_c = \frac{1}{5} S_p$

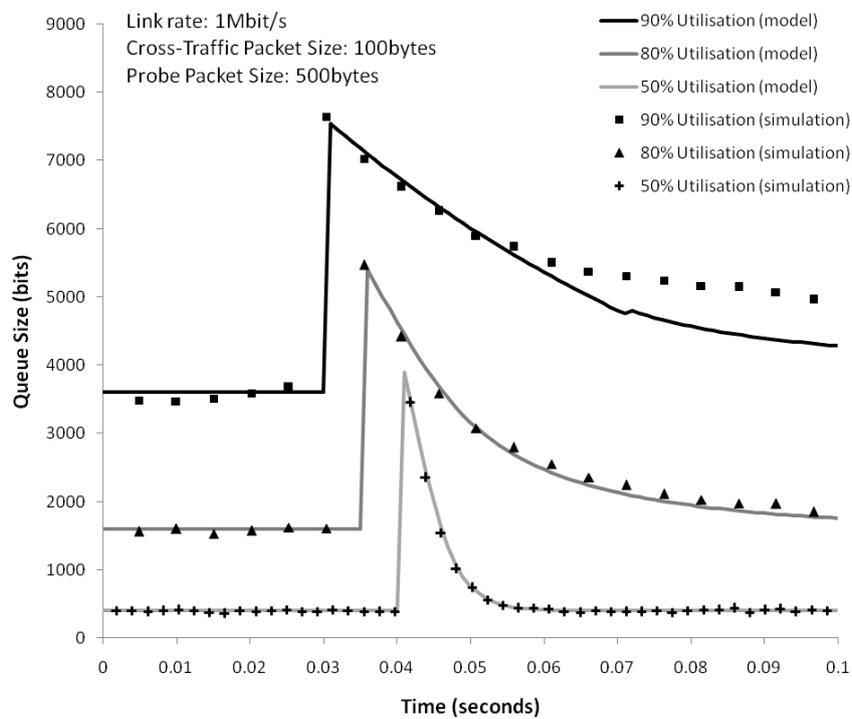
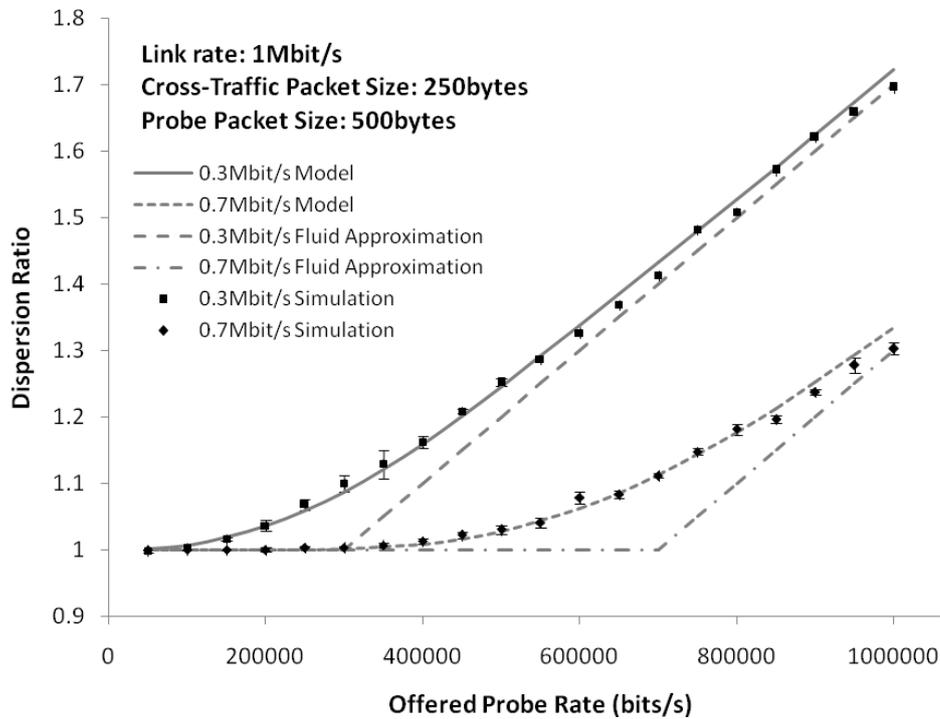


Figure 8: Simulated mean queue-size profiles observed during the passage of a 500byte probe packet, using (a) 250byte and (b) 100byte cross-traffic, compared with the model's predictions. (Each data-point represents the average of 1000 simulations.)

TOP OF FIGURE

(a) $S_c = \frac{1}{2} S_p$



(b) $S_c = \frac{1}{5} S_p$

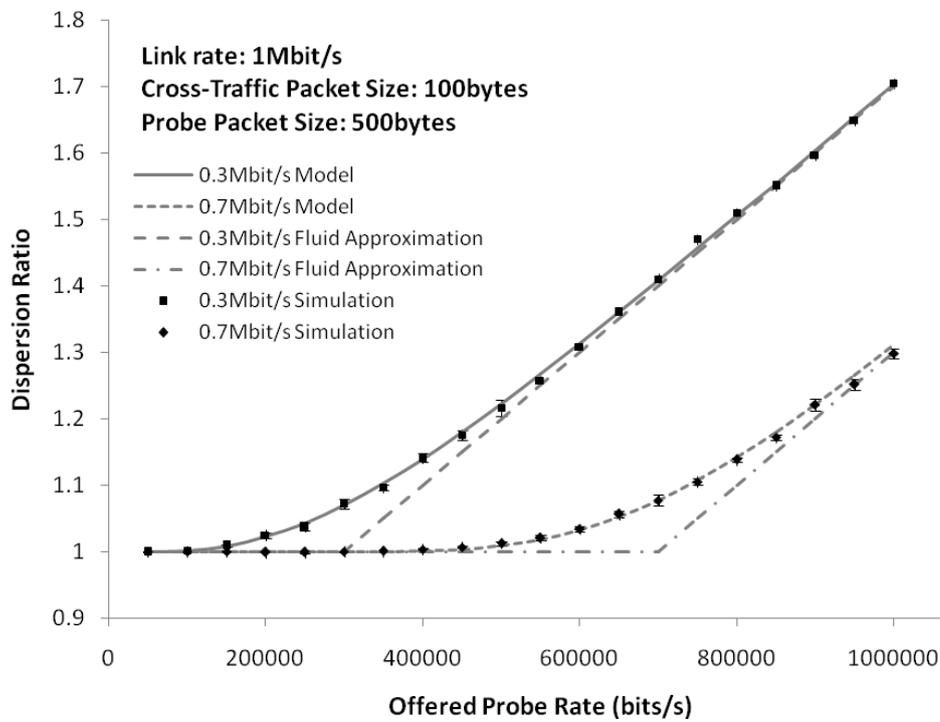


Figure 9: Comparison of analytical and simulated dispersion ratios using available bandwidth 0.3 and 0.7Mbit/s (70% and 30% utilisations respectively) and cross-traffic packets (a) one half and (b) one fifth the size of the probe packets. Each data point represents the average of five runs of 500 packet pairs each, and the error bars indicate ± 1.96 times the standard mean error (95% confidence interval).

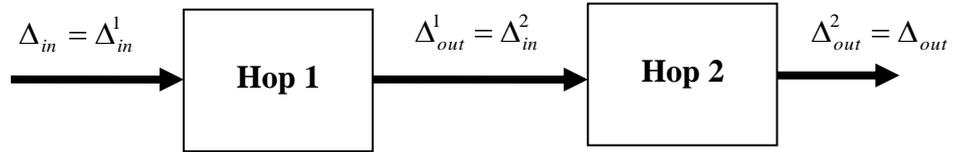
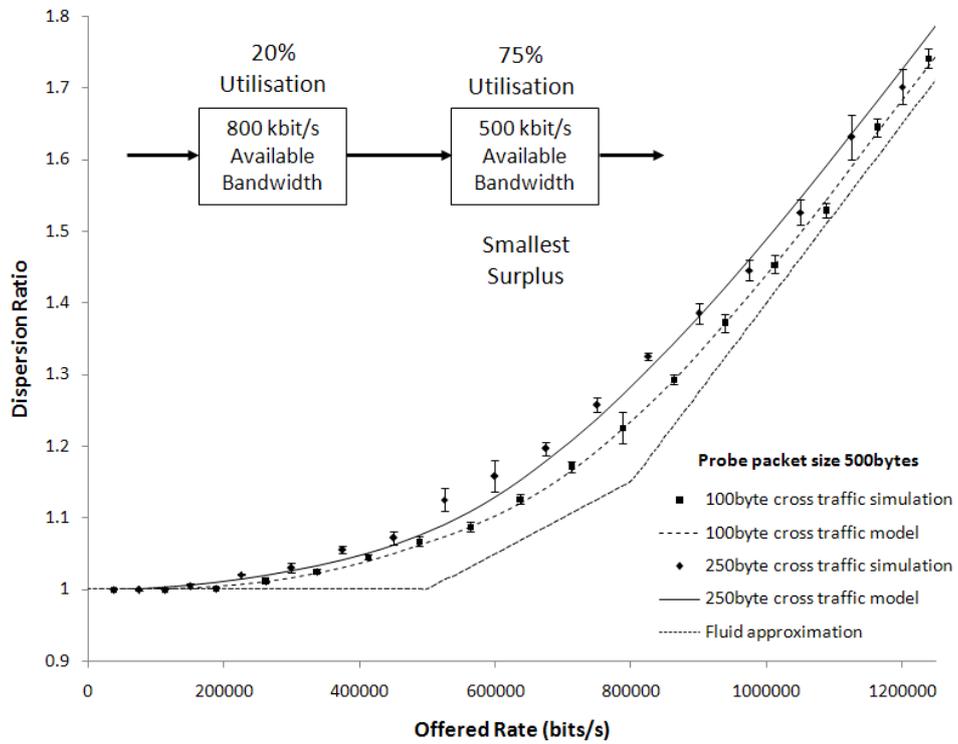


Figure 10: Iterative application of single-hop model in the analysis of a multiple-hop topology.

TOP OF FIGURE

(a) LSF Configuration



(b) SSF Configuration

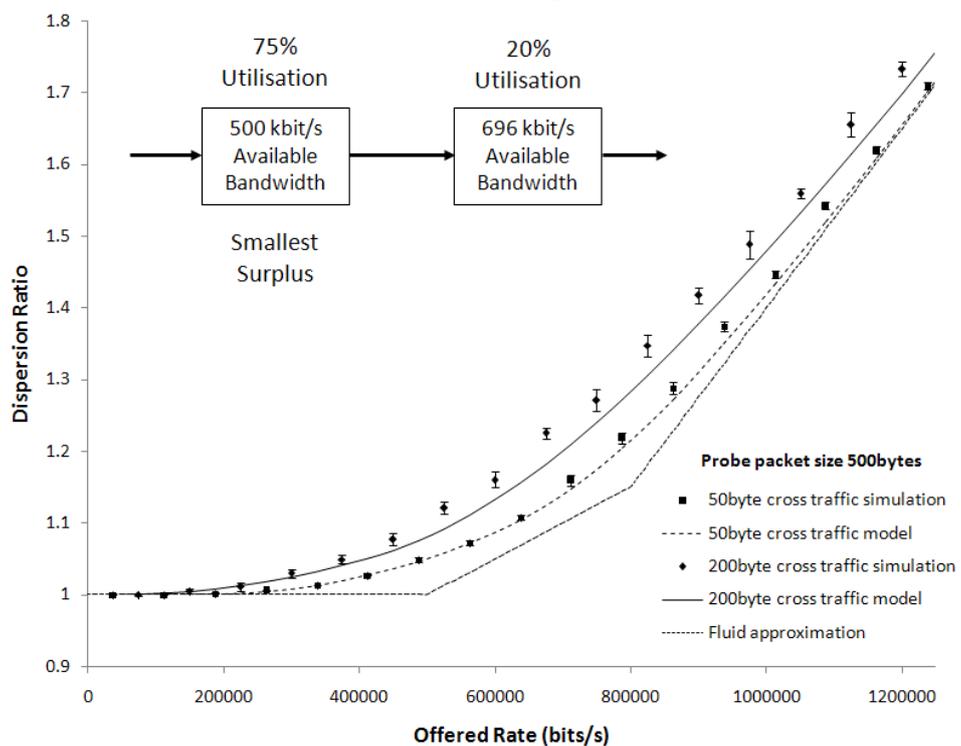


Figure 11: Simulated and analytical dispersion rate profiles obtained for a two-hop network path in (a) largest surplus first and (b) shortest surplus first configuration. Error bars indicate ± 1.96 times the standard mean error (95% confidence interval).

TOP OF FIGURE

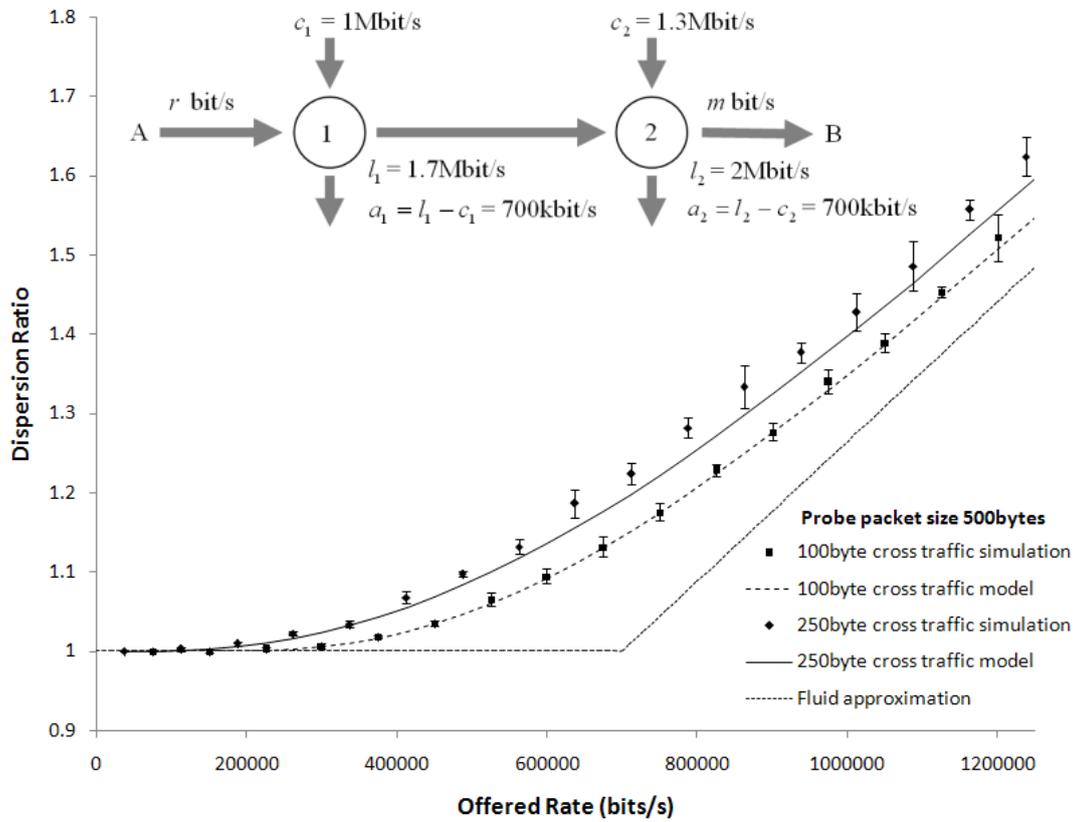


Figure 12: Comparison of analytical and simulated dispersion ratio profiles for a two-hop network path with equal available bandwidth at each hop (0.7Mbit/s). Error bars indicate ± 1.96 times the standard mean error (95% confidence interval).