

PathSensor: Towards Efficient Available Bandwidth Measurement

Péter Hága

*Department of Physics of Complex Systems, Eötvös University, Budapest, Hungary
and also at Collegium Budapest, Hungary*

E-mail: `haga@complex.elte.hu`

Attila Pásztor

Ericsson Research Hungary

E-mail: `Attila.Pasztor@ericsson.com`

Darryl Veitch

*Australian Research Council Special, Research Center for Ultra Broadband Information Networks,
Dept. of Electrical and Electronic Eng., University of Melbourne, Australia.*

E-mail: `dveitch@unimelb.edu.au`

István Csabai

*Department of Physics of Complex Systems, Eötvös University, Budapest, Hungary
and also at Collegium Budapest, Hungary*

E-mail: `csabai@complex.elte.hu`

Abstract

Efficient and reliable available bandwidth measurement remains an important goal for many applications. Recently a new class of active probing techniques has emerged which has clear advantages over earlier approaches, prominent members being pathload and pathChirp. They are based on observing an increased separation of probe packets due to local saturation of the queue at the narrow link, and do not require the knowledge of link capacities along the path. Each can deliver reasonable estimates of (average) available bandwidth, but has important drawbacks. Despite the shared general principle, there are many dimensions involved to fully define a given method in the class. We clearly delineate these, and point out unexplored regions of the ‘method design space’, and their importance. From these we define a configurable meta-tool, ‘PathSensor’, which supports methods within a promising new region, and whose modular structure allows different specific methods to be conveniently explored, and sensibly compared. We provide a compact simulation environment, PSIM, suitable for the analysis of a wide variety of active probing methods, and a realistic cross traffic generator. A specific method within PathSensor is selected and PSIM used to explore its performance. Comparisons are made against pathload and pathchirp under a variety

of cross traffic scenarios. Further comparisons are made using controlled non-stationary cross traffic in a network testbed. We show that PathSensor can exploit the low invasiveness of pathchirp with greater robustness and simplicity. Numerous directions for enhanced estimation accuracy are given.

1. Introduction

Efficient and reliable available bandwidth measurement is an important goal both for ongoing or on-demand monitoring of the state of the network, and for many applications which could use it to adapt their behaviour, for example to avoid congested paths.

By *available bandwidth* (AB) we refer to the instantaneous free transmission capacity of the route, where a route is seen as a series of *hops* in tandem. More precisely, the available bandwidth $\gamma_{\Delta}^h(T)$ of hop h over the time interval $[T, T + \Delta]$ is:

$$\gamma_{\Delta}^h(T) = \mu^h - \frac{q_{\Delta}^h(T)}{\Delta}, \quad (1)$$

where μ^h is the physical bandwidth of the link, and $q_{\Delta}^h(T)$ is the amount of traffic in bits that was served at the hop

during the time interval $[T, T + \Delta]$. Roughly speaking the available bandwidth is the difference of the link capacity and the average cross traffic (CT) rate. Based on this hop definition, we can define the available bandwidth of a *route*, in the time interval $[T, T + \Delta]$, as smallest hop AB on the path:

$$\gamma_{\Delta}(T) = \min_{h=1..H} \gamma_{\Delta}^h(T) = \min_{h=1..H} [\mu^h - \frac{q_{\Delta}^h(T)}{\Delta}], \quad (2)$$

where H is the number of hops. We readily acknowledge that this definition is not suitable for all purposes, for example flow control feedback can put into question the very meaning of ‘available’. However, we believe it is of value to be able to measure a quantity which describes path ‘state’ which is in some sense generic rather than application dependent.

Recently a new class of active probing techniques, dubbed the *interaction class* in [1], has emerged which has clear advantages over earlier approaches. The most important of these are:

- (i) not requiring a knowledge of the link bandwidths along the end-to-end path (although this may have seemed essential given the above definition), and
- (ii) not being based on assumptions about *which* link on the route is the *narrow link* (that with minimal hop AB).

The methods are based on observing an increased separation of probe packets due to local saturation of the queue at the narrow link, as illustrated in figures 1 and 4. This has been called *self-induced congestion* [3].

Two prominent members of this class are *pathload* [2] and *pathChirp* [3]. Each can deliver reasonable estimates of (average) available bandwidth, but each have drawbacks. *pathload*, the performance of which has now been quite well studied [4], suffers in particular from high invasiveness. *PathChirp* uses probes much more efficiently, but has complex and adhoc statistical analysis techniques, resulting in poor robustness. In our testbed measurements (in section 3.2) we used the same probing pattern as *pathChirp* with our new estimation method.

Our aim is to provide estimation methods which improve the state of the art significantly. We begin by taking due account of the different elements of an overall solution. These include: the design of the probe pattern itself, pattern parameters and sending schedule, the choice of space or time based aggregation, the measurement observable, choice of statistic to measure that observable, spatial and temporal smoothing techniques, threshold principles/heuristics, and finally estimate definition. We bear in mind that the performance of each of these is a function of cross traffic characteristics.

It is beyond the scope of this abstract to fully explore each of the above. What we do provide are arguments for why new choices should be made for some of the most im-

portant of them, thereby identifying a new ‘method design space’, within which promising new methods can be developed. This space is characterised, amongst other things, by using chirp-like probe patterns, and normalised *delay variation* rather than *delay* as the underlying observable. To support such development, we define the features of configurable meta-tool, *PathSensor*, which modularises design dimensions such as probe pattern definition, choice of statistic, smoothing approach, and detailed AB estimate definition. We aim to make *PathSensor* available to enhance the community’s ability to explore, and meaningfully compare within a common framework, not just different *methods* but different method *dimensions*. We use (and will also provide) a compact simulation environment, PSIM, suitable for the analysis of a wide variety of active probing methods, including a realistic cross traffic generator, to explore the principles underlying *PathSensor* and the impact of cross traffic variability.

To demonstrate the potential of this idea, we configure *PathSensor* to realise a specific method within the family. We make comparisons using controlled non-stationary cross traffic in a network testbed to test this method against *pathload* and *pathChirp* under a variety of cross traffic scenarios. In this paper we used Poisson and Weibull renewal cross traffic both in simulations and measurements. We also compare, in both the PSIM and testbed environments, two different methods within *PathSensor* which are identical except in the final heuristic defining the AB estimate. We show that *PathSensor* method(s) can exploit the low invasiveness of *pathChirp* with greater robustness and simplicity. The specific methods we select are still preliminary from many points of view, and it is expected that more work will identify refinements as well as new methods with better properties.

2. Principles of PathSensor

2.1. The Interaction Class, Methodology and Observables

The underlying principle of the interaction class is illustrated in figure 1 by using one-way delay, the same observable used by *pathload*. It is simply the fact that the queueing delay across a route increases steadily once the probing rate exceeds the available bandwidth, as the queue at the narrow hop is pushed into instability. In this simple 1-hop example (with Poisson CT), well spaced probe pairs are sent with inter-pair separations which decrease with every pair, defining local probing rates (LPR) which *increase*. The transition to instability is clear, and intuitively, the transition point represents an estimate of available bandwidth. This experiment, and others in this section, were conducted using PSIM described in section 3.1.

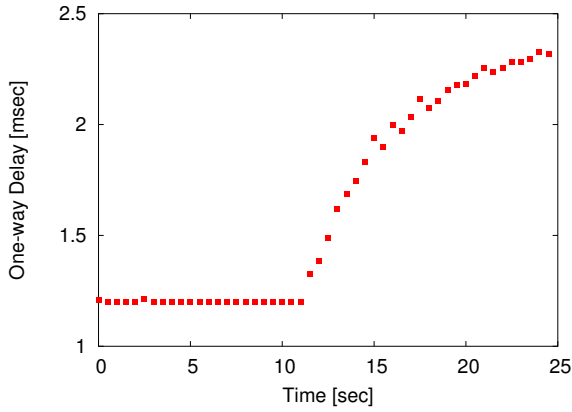


Figure 1. One-way delay as a function of probing rate. A steady increase marks the transition of the queue at the narrow hop from stable to unstable. Only a single probe per local probing rate gives unreliable estimates

Let us define the main properties of probing methods. Packet i arrives to the hop by appearing in the queue at the time instant τ_i . It begins service after a waiting time of $w_i \geq 0$, completes it after a service time of $x_i > 0$, and after a constant propagation delay of $D > 0$, exits the hop at time τ_i^* . The delay is therefore

$$d_i = \tau_i^* - \tau_i = w_i + x_i + D, \quad (3)$$

and comparing two adjacent packets we have the

$$\begin{aligned} \text{inter-arrival time: } t_i &= \tau_i - \tau_{i-1}, \\ \text{inter-departure time: } t_i^* &= \tau_i^* - \tau_{i-1}^*, \\ \text{delay variation: } \delta_i &= d_i - d_{i-1} = t_i^* - t_i \\ &= (x_i - x_{i-1}) + (w_i - w_{i-1}). \end{aligned} \quad (4)$$

The variability in the curve in figure 1 is a reflection of the variability in the CT encountered. In fact, different trials of this ‘continuously increasing probe rate’ experiment would produce different curves, and at fixed local probe rate we would observe an entire distribution of delay values. Two such are given in the first column of figure 2, one at a low LPR (top plot) and one below (bottom). The distribution has shifted to higher values after the AB was exceeded, but otherwise there was little qualitative difference.

An alternative way to examine the effect of changing LPR is to look directly at the growth $t_i^* - t_i$ of the inter-probe arrival separation t_i^* (measured from the last bit) at the receiver, compared to its original value t_i at the sender. In fact, this quantity is none other than the delay variation

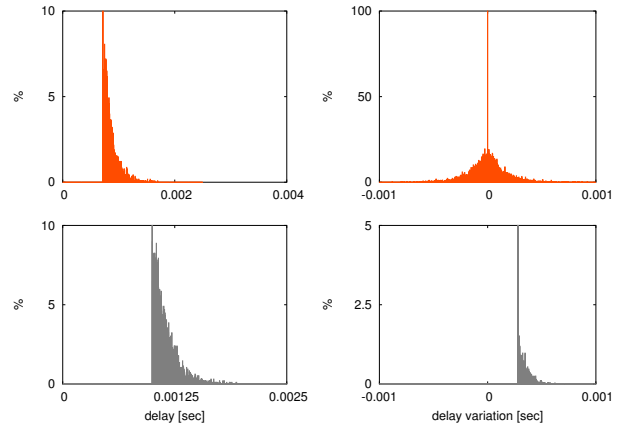


Figure 2. Histograms for delay (first column) and delay variation (second column) based on two experiments, each of 12,000 packet pairs at two fixed probing rates (those marked in figure 1): Top: low probing rate, stable case, Right: high probing rate, unstable case. The delay histograms show a shift but little else, whereas delay variation shows a profound change in shape.

$\delta_i = d_i - d_{i-1}$ between a probe pair. *PathChirp* is based on delay differences. As a result, it benefits from the advantages of a delay variation based description. These include immunity to clock synchronisation issues, and the concentration into highly localised discrete peaks, events characteristic of probes which share the same busy periods. We detail these and related issues in [6].

A further alternative way is to consider the *relative growth* t_i^*/t_i of the before and after probe separations. This gives us a dimensionless quantity where excursion from the limiting zero CT value of 1 are normalised in a natural way, enabling meaningful comparison. This new observable, which we call the *relative dispersion*, is simply related to the delay variation $\delta_i = d_i - d_{i-1}$ between a probe pair, via

$$s_i := \frac{t_i^*}{t_i} = \frac{\delta_i}{t_i} + 1. \quad (5)$$

and so also benefits from the advantages mentioned above.

The delay variation histograms corresponding to those on the left of figure 2 appear on the right in the same figure. They show a much clearer distinction between the characteristic before and after the transition point. The symmetric shape in the left hand plot (call the *independence signature* in [6]) is a reliable indicator that probes within pairs do not share the same busy period. In contrast, the highly asymmetric shape in the left plot shows that they do.

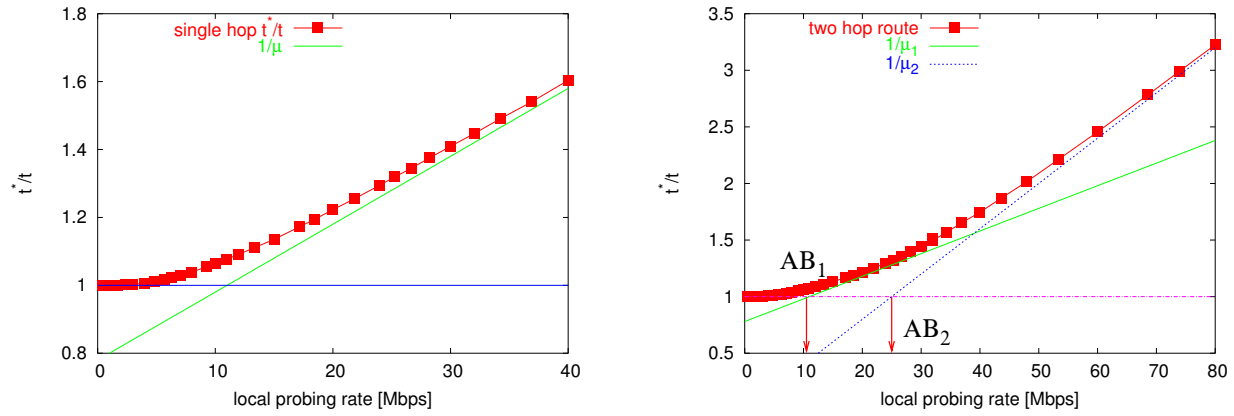


Figure 3. Expected system curves (using mean statistic) for Poisson CT. Left plot: single hop system, Right plot: two hop system, narrow link followed by bottleneck.

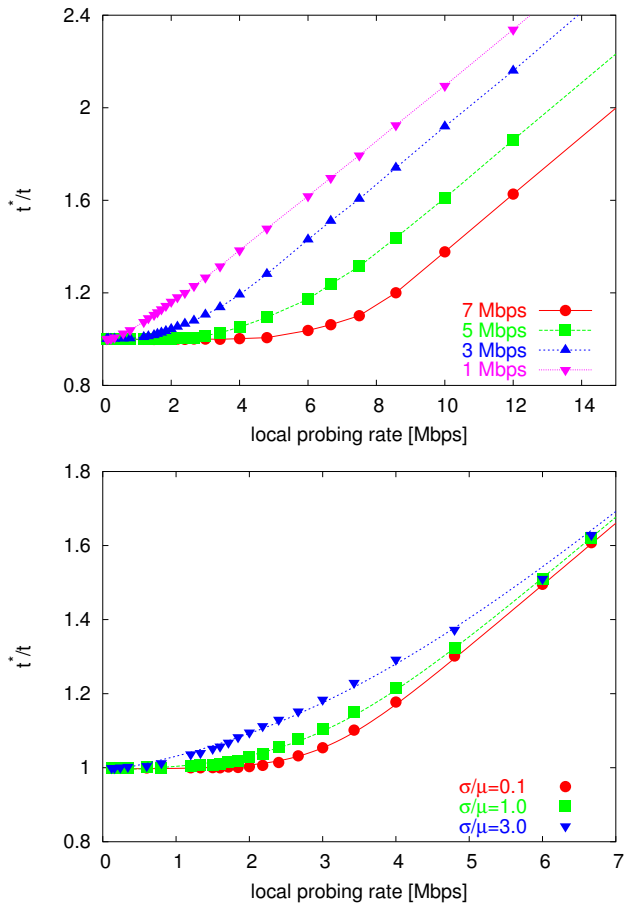


Figure 4. The curve of the t^*/t starts increasing at different probing rate for different mean of cross traffic, and the curve has different roundness for different variation of cross traffic.

The simulations used to produce figure 2 are **not** intended to emulate a practical estimation method. Instead, they are an important example of our methodology for meaningful evaluation of method behaviour. The packet pairs involved are well separated. Each effectively sees the same system in equilibrium, free of ‘self interference’ across different pairs. The resulting distributions thereby measure the true (CT dependent) response of the system in a way which is estimation method independent. From these distributions, we can choose a statistic to summarize the system response in the form of a function of LPR which can be thought of as the ‘true’ response of the path plus cross traffic.

If we select the mean as the statistic, then figure 3, left plot, is the result for the system displayed in figure 2. It is the ‘expected curve’ in a probabilistic sense, which estimation methods will effectively be trying to randomly sample, imperfectly, at a finite set of local probing rates. With this foundation, we are in a position to define methods and breakdown their performance in well defined dimensions. We do this in section 2.2 in the context of *PathSensor*. First, we demonstrate the utility of these system curves in two separate ways.

The examples given so far have been for a one-hop route. In that case, as seen in the left plot of figure 3 at large values of LPR the slope of the expected curve tends to a constant. It is not difficult to see that this slope is just the inverse of the service rate of the hop. In a multi-hop route, if the narrow hop and the bottleneck hop coincide, then this continues to hold. Otherwise however, the situation is more complex. We explain this in the right plot of figure 3, where we have calculated the expected system curve (mean based as before) for a 2 hop system. AB_1 signs the available bandwidth of the narrow (first) link, while AB_2 signs the available bandwidth of the bottleneck (second) link. As

the LPR increases, we again see a transition as the AB is exceeded (at AB_1), but it is difficult to see the slope corresponding to the narrow hop (μ_1), as the increasing LPR begins to saturate the following hop (at AB_2), which is the bottleneck link in this case. It is the second hop here whose rate determines the final slope of the expected curve (μ_2).

In figure 4 we explore the shape of system curves as a function of CT characteristics. The top plot is not unexpected: the transition point moved to the right as the mean CT rate decreases, corresponding to increased AB. The lower plot shows the non-linear effect of CT structure on queuing performance: as burstiness increases with increasing coefficient of variation, queuing causes the system curve to move up increasingly prematurely, before the point of loss of stability is actually reached. The curve also becomes rounder, and therefore the transition point more difficult to detect (the arrival process of the CT here is Weibullian-renewal)

The above effects, revealed through the availability of the system curves, are some of the factors that estimation methods will have to deal with, but with the added complication of noisy data derived from small numbers of measurements.

2.2. Principles and Dimensions of PathSensor

We first describe the *principles* of *PathSensor*, which define the scope of the methods included, and then the *dimensions*, which are the main factors one can play with to design specific methods.

We have already described the first defining principle:

Principle 1: the observable is relative dispersion, $\{s_i\}$.

As we have already seen, the observable of *pathload* is delay, and that of *pathChirp* delay variation.

The second principle relates to the probe pattern:

Principle 2: the fundamental probe pattern is a chirp.

Like *pathChirp*, we adopt a chirp-like signal (one with exponentially changing frequency) because of its ability to logarithmically scan a range of bandwidths in a single pattern. An example is provided in figures 5 and 6b. The distances between probes begins very large, and are decreased geometrically by a *spreading factor* ([3]), a : $t_j = at_{j-1}$. Initially the probes do not meet each other, but as the LPR p/t_j increases (where p is the probe size, they will) begin to self interfere rarely, and then severely.

The chirp pattern is not only very economical (especially compared to the probe pattern used by *pathload*) in terms of the number of probes, it also eliminates the need for external iteration and feedback as in *pathload* (figure 6a). Instead, the ‘iteration’ is done via the analysis algorithms, which can be varied at will, allowing for fast and on-line estimation.

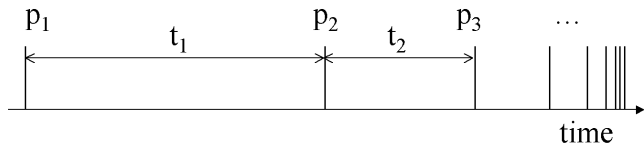


Figure 5. The pattern of packet in a chirp probe pattern.

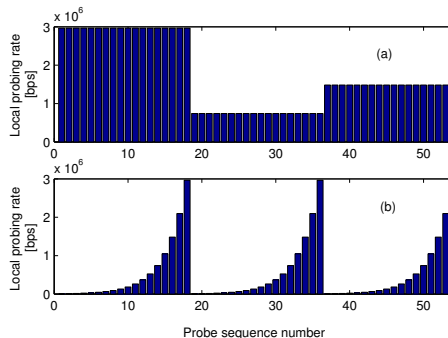


Figure 6. Probing patterns, three periods using each pattern are shown. The patterns are: (a) packet train with a controlled and constant rate within the train - used by NEPRI, TOPP, *pathload*; (b) chirp, a single pattern covering a whole interval - used by *pathChirp* and *pathSensor*.

Principle 3: the primary aggregation dimension is temporal.

To explain what this means, consider a measured system curve based on data collected from N identical chirps containing M discrete local probing rates t_j . There are many choices for how to combine these NM numbers. Principle 3 says that we must first summarize data for j fixed, that is temporally over the chirps with LPR fixed, before considering the spatial dimension. This philosophy is guided by the fact that a system curve estimated from a single chirp is extremely poorly sampled, so that inference based on it is likely to be very unreliable. In contrast, we can hope to benefit from the $1/N$ decrease in variance characteristic of independence by aggregating over the chirps for each j fixed, as chirps are widely spaced. *PathChirp* does not obey this principle, as it tries to extract information, and even AB estimates, from within a single chirp before applying any aggregation over time, and *pathload* inherently performs spatial processing first.

The first dimension is

Dimension 1: probe pattern parameters.

There are many possible parameters, apart from a and the initial and final probing rates, which should be supported, to enable probing streams which are considerably

richer than that considered in *pathChirp*, where identical chirps were sent periodically, with large gaps between them. These include:

- probe size p_j as a function of the probe 'serial number' j .
- j dependent spreading factor,
- chirps 'pairs' where parameters alternate,
- packets replaced by packet pairs or trains within a chirp,
- combined 'increasing' and 'decreasing' rate chirps.

To first order, we can think of the above as defining at which rates the service curve will be sampled at, and in which order.

Dimension 2: Summary statistic of the temporal aggregate.

This refers to what to do with the N measurements for a fixed probing rate j as defined above. Reasonable choices include the mean and the median.

Dimension 3: Type of spatial aggregation.

Given a statistic or statistics measured after temporal aggregation, there remains the choice of how to combine them spatially, that is over the probing rate dimension. As even the aggregated statistics can be highly variable, it is important to make maximum use of a priori knowledge of the system to increase robustness. For example we know that in the limit of small rates, that both the mean and median of $\{s_i\}$ are 1. Also, it is intuitively clear that the system curve should be monotonically increasing.

Dimension 4: Available Bandwidth Definition.

As we pointed out, the local probing rate at which the transition starts is clearly related to the available bandwidth. The exact nature of the change however greatly depends on the characteristics of the cross traffic and the probe stream, as demonstrated by the results of figure 4. The method to determine the transition point may involve heuristics of different kinds, informed by diverse scaling arguments.

3. Comparisons: Methodology and Infrastructure

We provide comparisons both in a totally controlled simulation environment, using our `PSIM` tool and CT generator, and a partially controlled network testbed environment, where the same probe traffic and CT traffic generators are used.

We chose to use the simulator to provide clear, well defined benchmark experiments using stationary CT, to

consider performance without the complication issues surrounding non-stationary versus variability of available bandwidth. In contrast, we use the testbed environment with controlled CT to test the robustness of the methods to true changes in AB, and their ability to track them.

3.1. PSIM

We developed a light-weight simulation package, `PSIM` [7], to help us efficiently investigate the main characteristics of different active probing methods. The main task of such tool is to generate information corresponding to what would be obtained during an active probing experiment. This means that there is no need to routing and transport protocol simulation, the required outputs can be calculated according to queueing equations which describe the delays along the route. The tool efficiently solves these equations numerically and is capable of handling input and output parameters in a convenient and effective way. For this purpose we have decided to implement `PSIM`, which we call a 'simulation' tool.

3.2. Network Testbed

A simple setup with 3 PCs and 1 router was used in our laboratory experiments. One PC was used as a CT generator, the second as a probe sender, and the third received both the probe and CT streams. A DAG3.5E capture card monitored the receiver's incoming traffic and served as the source of the reference information for comparing the different measurement methods.

4. Results

We first define two methods within *PathSensor*. To ease a comparison with *pathchirp*, we chose a periodic probing stream with very similar average characteristics: 2 chirps of 20 probes per second, with $p = 1500$ bytes, (spreading factor of $\sqrt{2}$), and an average rate of 500kbps. We take the mean (our temporal statistic) over chirps for each of 20 probing rates, to obtain a single estimated system curve sampled at 20 rate values.

For the parameter estimation, we wish to find a parametric fitting family $f(x)$ which should satisfy the following properties:

- $f(x) \rightarrow 1$ as probing rate $x \rightarrow 0$,
- smooth and monotone increasing,
- asymptotically linear for large x : $f(x) \rightarrow Ax + B$.

We describe such a family below.

Finally, we test two algorithms to select the transition scale, resulting in two methods for AB estimation:

- Threshold based: $\gamma_{th} = x : f(x) = s^*$ (we use $s^* = 1.1$),

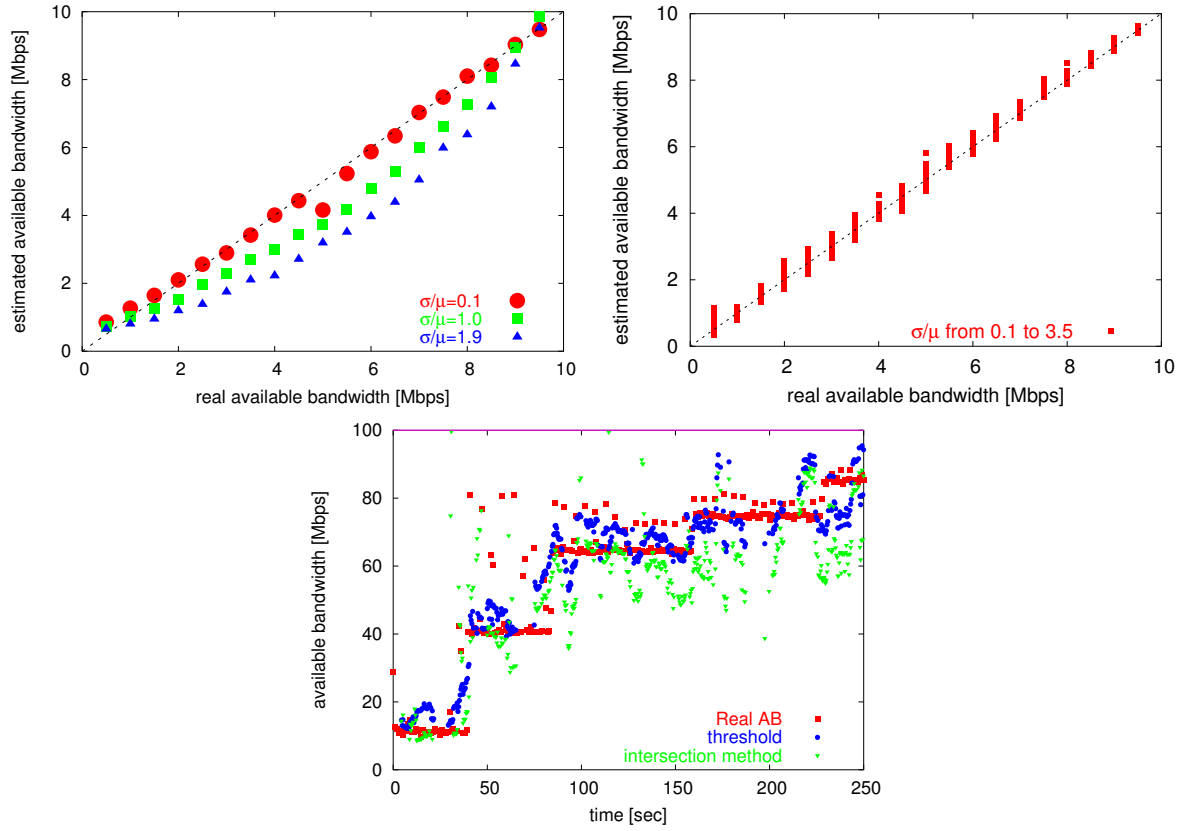


Figure 7. Comparing AB heuristics within PathSensor under stationary conditions. Left: performance of γ_{th} as a function of CT mean and burstiness. Right: γ_{int} scales more naturally to changing burstiness. Bottom: under non-stationary conditions, γ_{th} is more robust

(b) Intercept based: $\gamma_{int} = (1 - B)/A$, the intercept of the two asymptotes.

Such a family requires at least three parameters: two to fit the asymptotic slope, and a third to allow for the CT dependent speed of transition. The following family, which has a physical basis which we cannot enter into here, satisfies these requirements:

$$f(x; a, b, c) = \int a \cdot \frac{\tanh(b \cdot (x - c)) + 1}{2} dx \quad (6)$$

$$= \log \left((\cosh(a \cdot (x - c)))^{\frac{b}{2a}} \cdot \exp \frac{b \cdot x}{2} \right) + d,$$

where $d = 1 - bc/2 - b \log(2)/2a$.

One can show that $A = b$, and $B = 1 - bc$. Also, the intercept method has a nice functional form:

$$\gamma_{int} = c + \frac{3 \log 2}{2a}, \quad (7)$$

which reflects the natural location parameter role of c , and is independent of the final slope b . There is an other ad-

ditional benefit: the value of $1/b$ gives us the bottleneck physical bandwidth and the a parameter depends on the variance of the background traffic.

We determine the parameters $(a, b, c) = (a^*, b^*, c^*)$ of best fit (minimizing the χ^2 function via all the parameters) using the *Levenberg-Marquardt Algorithm*, a standard non-linear least squares(NLLS) method [8]. The fits were generally good both for Poisson and Weibullian-renewal CT, with some stability problems in some cases, and minor ad-hoc fixes required. In future work we will thoroughly reexamine the minimisation algorithm to eliminate these problems.

4.1. Comparisons

We first made a comparison between the two *PathSensor* methods in simulation under stationary conditions. We simulated to cover a large parameter space with average cross traffic rates, and coefficient of variation (σ/μ) values. The results comparing the performance of γ_{th} and γ_{int}

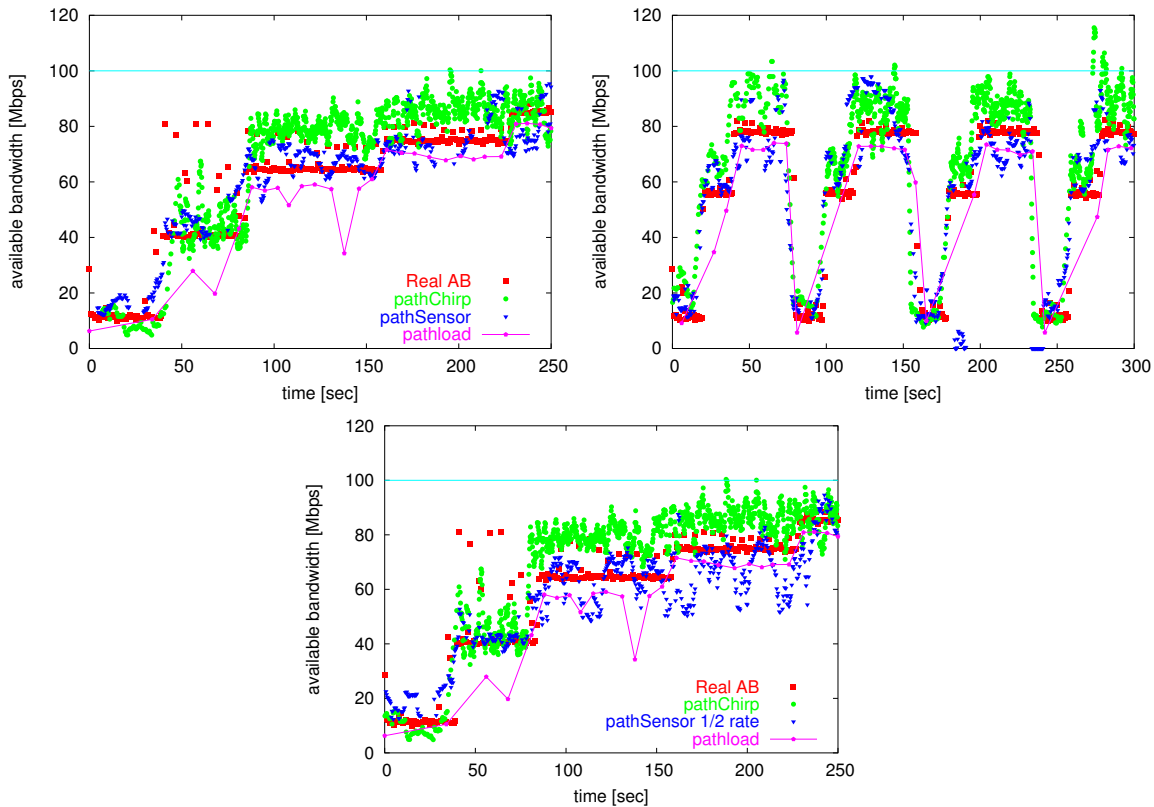


Figure 8. Comparison of *pathload*, *pathChirp* and *PathSensor* methods in a real 2 hop laboratory network. Top: slowly increasing CT, *PathSensor* is more robust, Middle: rapidly changing CT, *pathload* is too slow when AB is low Bottom: *PathSensor* remains comparable to *pathChirp* even with half as many probes.

and shown in figure 7. In the left graph, results for γ_{th} begin well but worsen as burstiness increases. This is not surprising, as the idea of a threshold is ad-hoc and would not be expected to scale with CT properties. In the central plot, γ_{int} is much less sensitive to burstiness level, which make intuitive sense, since as the system curve becomes rounder due to increased burstiness, the intercept naturally moves to the left. In the right plot however the performance of the methods undergoes a transition as the AB changes. The intercept method seems to be too sensitive to noise if the transition is abrupt, as it is in the right of the plot for high AB.

Both method has some uncertainty (under estimation and non convergent fit). To minimize the error we choose always the smaller value of the two estimates.

In figure 8 we test the threshold method against *pathload* and *pathChirp*, using controlled non-stationary traffic in the laboratory. In each plot we see that *pathSensor* and *pathChirp* are roughly comparable, with *pathSensor* being a little more stable. Estimates from *pathload* are also

fairly comparable in the top left plot, where the AB changes slowly. In the top right plot however, it is clear that it cannot follow the rapid changes, in particular for low values of AB, where the iterative technique takes longer to arrive at an estimate. Finally in the bottom plot we half the number of points used by *pathSensor* by increasing the spreading rate. Despite this, the results remain comparable with those from *pathChirp*.

In the case of unloaded link our estimation is much better then the estimation with *pathChirp*. In the case when the fluctuating of the cross traffic is rapid the two methods gives us almost the same result.

5. Conclusion

We have made the first steps to work out the underlying theoretical model for evaluation of the measured data. We have presented an improved method for available bandwidth estimation and we have shown its capabilities in both simulation environment and real network measurements. In the future we would like to develop a complete mea-

surement tool which would include different, changeable modules for the different estimation techniques.

6. ACKNOWLEDGEMENTS

The authors acknowledge the support of Ericsson. Darryl Veitch acknowledges the support of the Australian Research Council. Péter HÁga and István Csabai acknowledge the support of European Union IST FET Integrated Project EVERGROW and the support of the NKFP 2/032/2004 project.

References

- [1] Attila Pásztor, Darryl Veitch, and Péter HÁga, "On evaluating a new class of available bandwidth methods," in *ISMA First Bandwidth Estimation Workshop*, San Diego, CA, 9–10 Dec 2003.
- [2] Manish Jain and Constantinos Dovrolis, "End-to-End Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput," in *Proceedings of ACM SIGCOMM'02*, Pittsburgh, Pennsylvania, Aug 19-23 2002.
- [3] V. Ribeiro, R. Riedi, J. Navratil, and L. Cottrell, "pathChirp: Efficient Available Bandwidth Estimation for Network Paths," in *PAM 2003, Passive and Active Measurement Workshop*, La Jolla, California, April 6–8 2003.
- [4] R. Prasad, M. Jain, and C. Dovrolis, "Evaluating pathrate and pathload with realistic cross-traffic," in *ISMA First Bandwidth Estimation Workshop*, San Diego, CA, 9–10 Dec 2003.
- [5] C. Dovrolis, P. Ramanathan, and D. Moore, "What do packet dispersion techniques measure?," in *Proceedings of IEEE Infocom'01*, Anchorage, Alaska, April 22–26 2001.
- [6] Attila Pásztor and Darryl Veitch, "On the scope of end-to-end probing methods," *IEEE Communications Letters*, vol. 6, no. 11, pp. 50–511, Nov. 2002.
- [7] Attila Pásztor, *Accurate Active Measurement in the Internet and its Applications*, Ph.D. thesis, University of Melbourne, Victoria 3010, Australia, 2003.
- [8] "Numerical Recipes: The Art of Scientific Computing," Cambridge University Press, <http://www.nr.com/>.
- [9] "Endace Measurement Systems," <http://www.endace.com/>.