End-to-End Inference of Link Level Queueing Delay Statistics

Conference Paper · January 2010

DOI: 10.1109/GLOCOM.2009.5425672 · Source: IEEE Xplore

6 authors, including:

Gianni Antichi
University of Cambridge

Domenico Ficara
Università di Pisa

S. Giordano
Università di Pisa

Fabio Vitucci
Università di Pisa

End–to–End Inference of Link Level Queueing Delay Statistics

Gianni Antichi, Andrea Di Pietro, Domenico Ficara, Stefano Giordano, Gregorio Procissi, Fabio Vitucci
Dept. of Information Engineering, University of Pisa
Via Caruso 16, 56122 Pisa, Italy
Email: <first.last> @iet.unipi.it

***Abstract*—Characterizing delay distribution over the links of a network provides a remarkable amount of information which can be useful for troubleshooting, traffic engineering, adaptive multimedia flow coding, overlay network design, etc. Since querying each and every node of a path in order to retrieve this kind of information can be unfeasible or just too resource demanding, the recent research trend is to infer the internal state of a network by means of end–to–end measurements. Many algorithms in literature require active measurements and are based on a single-sender multiple-receivers scheme, thus relying on the cooperation of a possibly wide number of nodes, which is a quite strong assumption. Moreover, many previous works adopt Expectation-Maximization algorithms to cope with large and under-determined equation systems, thus increasing the uncertainty of the final delay estimation. This paper, instead, proposes a technique to infer the cumulants of the delay distribution over each link of a given network path, based on two–points measurements only. The cumulants, in turn, can be used to approximate the distribution function through the Edgeworth series. The results of our approach are assessed through a wide series of model-based and ns2 based simulations and show fairly good performance under different network load conditions.**

I. Introduction

The statistical characteristics of the queueing delay across a computer network provide a remarkable amount of information that can be useful to many purposes such as network troubleshooting, overlay network design, managing delay sensitive applications etc. Several tomographic methods have been developed in order to infer the statistical characteristics of the queueing delay across a computer network, based on end–to–end measurements (see next Section II). Such a knowledge is very handy for delay sensitive applications (such as multimedia services), which may choose a particular server on the basis of the delay associated with the corresponding network path, as well as for network management and traffic engineering issues: a link introducing a heavy queueing delay is likely to be congested, and the network administrator, either manually or by using automated load balancing algorithms, can reroute incoming traffic flows away from that link.

Almost all the proposed tomographic techniques we refer to are intended to infer the delay statistics over a whole network by making use of active measurements performed by a probe sender node and a set of probe receivers. The network topology is thus modelled as a tree, whose root represents the probe sender and whose leaves stand for the receivers: for each arc of such a tree, a delay distribution is inferred. Unfortunately, very few of the proposed techniques (such as our previous work [2]) allow, in general, the inference of the statistics of the queueing delay on each physical link of a network path, since arcs of the topology tree are logical links that can be associated with a multitude of physical links. The delay distribution referring to a logical link can thus be considered as the convolution of the delay distributions referring to different physical links. In addition, the cooperation of several receiver nodes is not always available, thus reducing the extent of application of such algorithms. On the contrary, the solution that we propose is based on two–points measurements only, thus guaranteeing the maximum flexibility of use. Furthermore, most of the proposed approaches rely on complex estimation methods such as the Expectation-Maximization (EM) algorithm, which, as discussed in [2], can raise convergence problems and requires an undetermined number of iterations to converge.

With respect to state of the art techniques, our approach presents at least two novel contributions:

- it allows to infer information about the queueing delay of each link of an end–to–end path (including the queues of the reverse path) based on two-points measurements only;
- it allows to estimate the cumulants of the queueing delay distribution as a simple linear combination of the cumulants of the available data set (which, in turn, can be estimated without bias by means of the well known *k-statistics* [5]). The estimator is then unbiased and of low complexity (the coefficients of the linear combination are fixed and can be easily pre-computed offline).

The only limitation of our approach is the impossibility of calculating the first order cumulants of the queueing delays, since, in that case, the problem is intrinsically undetermined. However, we propose here an approach that allows to estimate such a statistic for the most congested links, which are the most relevant for traffic engineering and troubleshooting purposes.

To motivate the application of our algorithm, one may consider the possibility of identifying and isolating subsets of links of a complete network path which are responsible for end-to-end performance degradation, even in the case the path spans multiple domain. Based on that, several actions to cope with the problem can be undertaken, both in terms of traffic engineering choices (traffic can be routed away from

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

congested links by leveraging MPLS label switching) and administrative actions (in case the links belong to a provider domain, enforcement of SLAs can be requested). In this context, we point out that, typically, active delay measurements are often used for the purpose of SLA monitoring.

In a different scenario, the use of algorithms that infer information about link congestion status could be included into the routing metrics.

The rest of the paper is organized as follows: section II gives an overview of the currently available delay estimation techniques. Section III describes the main concepts of our proposed technique, both by describing our packet pair probing technique and by providing a detailed mathematical description of our estimator. Section IV describes the heuristic solution we propose for coping with the underdetermined problem associated to the inference of the mean queueing delay values. Section V assesses the accuracy of our estimation techniques through both model based and ns2 based simulations. Final remarks conclude the paper.

II. Tomographic techniques for queueing delay distribution estimation

Most of the algorithms which have been proposed in the literature are based on a discretized delay model: the domain of the possible delay values for each link is partitioned into a finite set of bins (which can be of either variable or fixed length) and the probability of the queueing delay falling within each bin is inferred. The inference of a continuous distribution is therefore transformed into a parameter estimation problem. However, different kinds of approaches can be also used: paper [3] proposes to model the queueing delay as a linear combination of different continuous distributions and to estimate the coefficients of such a mixture, while paper [4] proposes to estimate the cumulant generating function of the delay distribution. Such an approach takes advantage of the cumulant generating function of the end–to–end delay evaluated as the sum of the cumulant generating functions of the delay distributions over each link: a linear system involving end–to–end measurements and the unknown link level distributions can therefore be written. The measurements probes can be either multicast packets directed to the whole set of receivers (as proposed in [6]) or couples of unicast packets directed to a receivers pair. In order to provide a benchmark for our investigation, we will briefly describe here the basic concepts which are the basis of the state-of-the-art delay estimation techniques.

A unicast probe is composed by a pair of back–to–back packets directed to different receivers; once the packets reach their destination, their one way delay is measured. Such a delay is modelled as the sum of two random variables: the first one (referred to as d_s) represents the delay experienced on the shared path, and is assumed to have the same value for both packets (since the two packets are expected to cross the shared path back–to–back, perfect correlation is hypothesized), while the second one (referred to as y_s) represents the delay experienced on the path between the branching point and the receiver. For each packet, such a delay is assumed to be an independent random variable. More formally, let us consider a pair of probe packets directed to receivers i and j and let us indicate with Y_i and Y_j the corresponding end–to–end delays: the following relation holds

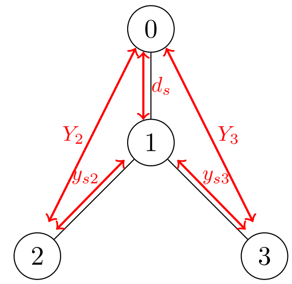

Figure 1. End–to–end and logical link level delays experienced by a packet pair probe.

$$\begin{array}{rcl} Y_i & = & d_s + y_{si} \\ Y_j & = & d_s + y_{sj} \end{array} \tag{1}$$

The relationship between the quantities involved is graphically represented in figure 1.

The vector of samples $\mathbf{Y}_{i,j}$, composed by the measurements pair $\{Y_i, Y_j\}$, is the basic data sample which is used by all unicast based estimation algorithms. The delay distribution inference is based on the knowledge of a certain number (N) of delay measurements $\mathbf{Y}_{i,j}^{(n)}$ for different pairs i, j of receivers. Since the information provided by those samples is insufficient for the purpose of a deterministic inference of the actual experienced delay values (the corresponding linear system would be widely underdetermined), the most common approach to the inference of the delay distribution is based on the Expectation Maximization (EM) algorithm [14]. In [2], we proposed the adaptation of a similar techniques to an end-to-end path, by taking advantage of the concept of virtual tree. However, our preceeding techniques showed several limitations, both from the point of view of numerical stability and in terms of processing complexity; therefore, in the following section we will present an improved approach that is both more reliable and less complex.

III. Link level delay distribution inference

The solutions described in the previous section are intended to provide estimates of the queueing delay distributions in a single–sender/multiple–receivers measurement scenario; however, in many cases, only two-points measurements are available. In general, an end–to–end path can be modelled as a sequence of links, each of them consisting of two independent queues, the one belonging to the forward path and associated with the queueing delay $X_f^{(k)}$ (k is the index of the node) and the one associated with the reverse path and corresponding to the queueing delay $X_r^{(k)}$; such a reference scenario is shown in figure 2. In this scenario, our main goal is estimating the cumulants of the delay distributions associated with each link of the path.

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

The $r-th$ cumulant of random variable X is defined as:

$$K_X^r = \left.\frac{\partial^r G_X(t)}{\partial t^r}\right|_{t=0} \quad (2)$$

where $G_X(t)$ is the cumulant generating function associated with X, defined as

$$G_X(t) = \log(\mathbb{E}(e^{tx})) \quad (3)$$

The reasons for the choice of such statistics are manyfold:

- the cumulants are linear with respect to the sum: the k-th cumulant of the sum of independent random variables is equal to the sum of the k-th cumulants of each random variable;
- unbiased estimators of the cumulants of a distribution are available through the use of k-statistics;
- the moments of a random variable can be exactly calculated based on the knowledge of its cumulants;
- the distribution of a random variable can be approximated based on the knowledge of the cumulants by means of either the saddlepoint method or the Edgeworth series.

Another interesting property of the cumulants, which will be crucial to the development of our technique, is:

$$K_{\alpha X}^r = \alpha^r K_X^r \quad (4)$$

which can be trivially derivated from (2).

We probe the end-to-end path by using a proper packet pair, whose property is already described in [2]. Such a probe is a packet pair, but, while the first packet is a unicast packet directed to the host at the other end of the path, the second is a *ping-like* packet (i.e. a packet that forces the receiver to send an immediate response to the sender, such as an *ICMP echo request*) directed to one of the intermediate nodes of the path, as shown in figure 2. The two packets are sent back–to–back by the probe sender and, as in the multiple receivers scenario, perfect correlation on the shared links is assumed; the generic n-th probe ($1 \leq n \leq N$) originates a pair of samples $\mathbf{Y}^{(l)}(n) = \{y_o(n), y_p^{(l)}(n)\}$, where y_o is the end–to–end delay experienced by the unicast packet directed to the other end of the path and $y_p^{(l)}$ is the round trip delay experienced by the $ping-like$ packet directed to the $l-th$ node of the path. By considering the path model previously described and by assuming the whole path consists of N_h links, the two quantities can be expressed as:

$$y_o(n) = \sum_{k=1}^{N_h} x_f^{(k)}(n) \quad (5)$$

$$y_p^{(l)}(n) = \sum_{k=1}^{l} x_f^{(k)}(n) + \sum_{k=1}^{l} x_r^{(k)}(n) \quad (6)$$

Such expressions can be verified by examining figure 2, where the behavior of one of the probes is illustrated; the two arrows indicate the path that is traversed by both the $ping-like$ packet and the one way packet, respectively. As we are intersted in the queueing delay only, a minimum filtering over the observed data is preliminarly performed in order to compensate for constant delay terms (e.g. transmission and propagation latencies).

Since the goal of our technique is to estimate the $r-th$ cumulants $K_{X^{(k)}_f}^r, K_{X^{(k)}_r}^r \forall\, k \in [0, N_{hop}]$ for any arbitrary order r, we can now write down a linear system which allows to compute such cumulants from the cumulants of the measurements which can be obtained through packet pair probing. A first set of equations can be easily obtained by re-writing (5) and (6) in terms of cumulants and by taking advantage of their linear property. Thus, the following equations hold:

$$\sum_{k=1}^{N_h} K_{x_f^{(k)}}^r = K_{y_o}^r \quad (7)$$

$$\sum_{k=1}^{l} K_{x_f^{(k)}}^r + \sum_{k=1}^{l} K_{x_r^{(k)}}^r = K_{y_p^{(l)}}^r \forall\, l \in [1, N_h] \quad (8)$$

The two relations above provide N_h+1 linearly independent equations; therefore, for the system to be solved, $N_h - 1$ additional independent relations are needed. In order to obtain them, let us consider the sum $S^(l)$ of the measured delays experienced by the two packets of each packet pair (i.e. the sum of the round–trip–time experienced by the ping-like packet directed to node l and the end–to–end delay experienced by the one-way packet). By combining (5) and (6), the following relation can be obtained:

$$S^{(l)}(n) = 2\sum_{k=1}^{l} x_f^{(k)}(n) + \sum_{k=1}^{l} x_r^{(k)}(n) + \sum_{k=l+1}^{N_h} x_f^{(k)}(n) \quad (9)$$

By expressing the relation above in terms of cumulants (to this end, let us take advantage of (4)), we can obtain $N_h - 1$ equations of the form ($\forall\, l \in [1, N_h - 1]$):

$$2^r \sum_{k=1}^{l} K_{x_f^{(k)}}^r + \sum_{k=1}^{l} K_{x_r^{(k)}}^r + \sum_{k=l+1}^{N} K_{x_f^{(k)}}^r = K_{S^{(l)}}^r \quad (10)$$

The overall linear system, for each cumulant order r, can then be written as:

$$\mathbf{H}^{(r)}\,\mathbf{X}^{(r)} \;=\; \mathbf{Y}^{(r)} \quad (11)$$

where $\mathbf{X}^{(r)}$ is the unknowns' vector:

$$\mathbf{X}^{(r)} = \left(X_f^{(1)}, \ldots, X_f^{(N_h)}, X_r^{(1)}, \ldots, X_r^{(N_h)}\right)^T$$

and the matrix $H^{(r)}$ is of the form:

$$\mathbf{H}^{(r)} = \begin{pmatrix}
1 & 0 & 0 & 0 & \ldots & 1 & 0 & 0 & 0 & \ldots \\
1 & 1 & 0 & 0 & \ldots & 1 & 1 & 0 & 0 & \ldots \\
1 & 1 & 1 & 0 & \ldots & 1 & 1 & 1 & 0 & \ldots \\
1 & 1 & 1 & 1 & \ldots & 1 & 1 & 1 & 1 & \ldots \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \ldots \\
2^r & 1 & 1 & 1 & \ldots & 1 & 1 & 0 & 0 & \ldots \\
2^r & 2^r & 1 & 1 & \ldots & 1 & 1 & 0 & 0 & \ldots \\
2^r & 2^r & 2^r & 1 & \ldots & 1 & 1 & 1 & 0 & \ldots \\
2^r & 2^r & 2^r & 2^r & \ldots & 1 & 1 & 1 & 1 & \ldots \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots
\end{pmatrix}$$

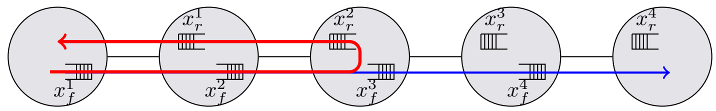

Figure 2. Working principle of the packet pair probes used for link-level delay estimation.

The vector of the known terms, in turn, is of the form:

$$\mathbf{Y}^{(r)} = \left(K^r_{Y_p^{(1)}}, \ldots, K^r_{Y_p^{(N_h)}}, K^r_{Y_o}, K^r_{S_p^{(1)}}, \ldots, K^r_{S_p^{(N_h-1)}}\right)^T$$

All of the terms in such a vector can be estimated from the data-set obtained by packet-pair probing, by means of non-biased estimators. It is easy to prove that $H^{(r)}$ is non-singular for each cumulant order $r \geq 2$: as a consequence, our technique allows to obtain an unbiased estimator for each of the cumulants of the link-level delay distributions except for the first order cumulant. Since the first order cumulant corresponds to the mean of a random variable, this conclusion is not surprising. Indeed, due to the linearity of the expectation operator, the problem of evaluating the average delay associated with the internal links based on the end–to–end measurements is formally equivalent to computing the value of each delay realization $X^{(l)}_{r,f}(m)$ which is, obviously, an undetermined problem. In the following section we will illustrate a heuristic method that allows to estimate the first order cumulant at least for the most congested links.

A final remark on the estimators' robustness is necessary: in the present work we do not actually provide theoretical confidence bounds for our estimates. However, we point out that the estimates yelded by our method are computed as linear combinations of the cumulants of the measured delays; the coefficients of such combinations only depend on the number of links and are therefore known a priori and the confidence ranges of the cumulants of experimental data-sets can be expressed analytically. Though fairly complex, theoretical confidence bounds can therefore be derived, but, for lack of space, we decide to omit such a discussion here and to postpone it to a follow-up work.

IV. Heuristic solution

As proved in the previous section, calculating the mean value of the link level delays based on end-to-end measurements only is an underdetermined problem. However, it is possible to extract from the available data-set some useful information concerning, at least, the most congested links. To this end, we will rely on the following hypothesis: in real networks there are few links where a packet experiences a heavy queueing delay, while the queueing delay on most links is often negligible; we will assume the mean delay associated to such links to be zero. If the links with negligible queueing delay could be located, the number of unknowns of the linear system could be reduced and hopefully, it would be possible to calculate the mean delay associated with the most congested links. Those lightly loaded links can be revealed based on the other cumulants: since their associated delay distributions can be approximated as a Dirac delta function, all of their estimated cumulants will be negligible. Once a lightly loaded link has been located, its associated column can be set to zero within the $\mathbf{H}^{(r)}$ matrix. Such a procedure can be iterated until the rank of the system reaches the number of non-negligible delay terms; when such condition is satisfied, the resulting overdetermined system (the number of equations is the same as that of the orginal system, but the number of unknowns has been reduced) can be solved by using, for example, a least squares approach. Such a technique would not work in the case of a link associated with a nearly constant but non-zero queueing delay (the distribution would in that case be modeled as a translated delta, whose translation factor could not be estimated throughthe cumulants). However, in this case, such a delay would be included in the constant delay slack which is compensated for during the preliminary minimum filtering phase and it would not be possible, in any case, to recover it from the available data-set. Of course, the number of delay terms that can be estimated depends on the specific delay distribution: deletion of a column in the system matrix can cause the overall rank either to be decremented or not, depending on the values in the specific column.

V. Experimental results

In order to assess the accuracy of our estimation algorithm, we performed two separate evaluation trials, one focused on the heuristic we described in the previous section and one focused on the estimation of higher order cumulants by means of the system described in section III. As for the assessment of the cumulant estimation accuracy, we ran simulations by using both Matlab generated random variables and Network Simulator ns2 testbed scenario. In the first case, we simulated a network path composed of 8 hops, each one associated with an exponentially distributed queueing delay. For the sake of clarity, we show the results associated only with the forward links (the estimator of the delay associated with the reverse links has the same mathematical form). The estimated and actual values of the cumulants up to the 6-th order are shown in figure 3.

As it appears from the graph, our estimates are in the large majority of cases very close to the actual value of the cumulants. Of course, the estimation error grows with the order of the cumulant; this is due to the fact that the variance of the k-statistics (the unbiased estimators used to retrieve the cumulants of the end-to-end measurement from the data-set) increases with the cumulant order. Therefore, since our estimator is a linear combination of such k-statistics, its variance grows as well, thus originating higher estimation errors. However, because of the consistency property of the k-

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

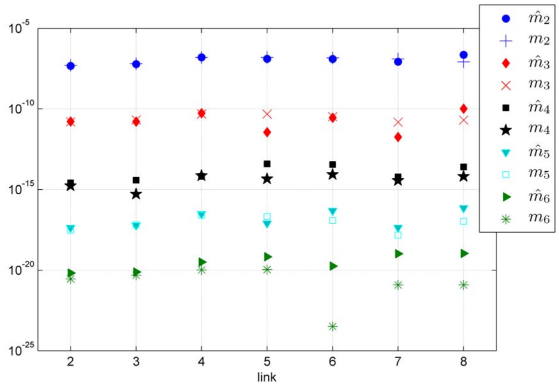

Figure 3. Estimated and actual cumulants of link delay distribution in a Matlab simulated model based scenario.

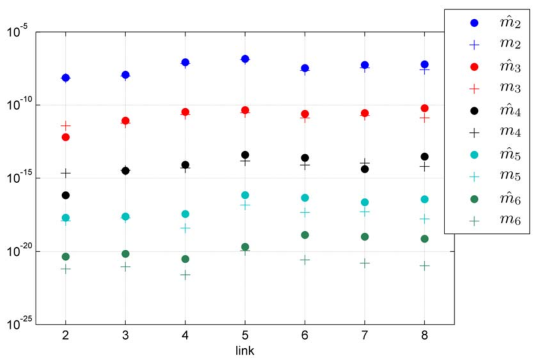

Figure 4. Estimated and actual cumulants of link delay distribution in a light-load scenario.

statistics, their variance can be arbitrarily reduced by using a larger data-set. As for the ns2 simulations, we set up a scenario composed by 8 links, each of them loaded with TCP cross traffic at a different rate; the TCP cross traffic crossing each link is generated by several connections, each of them characterized by a different segment size, in order to reproduce the multimodal distribution of packet lengths over the real Internet. Such a cross traffic is generated by out–of–path nodes and directed to different intermediate nodes of the path; the link connecting each out–of–path node with a node belonging to the path represents the bottleneck link of the TCP connections: by varying its capacity it is then possible to accurately tune the amount of TCP cross traffic loading each link of the path. Notice, however, that no assumption on assumption on the traffic distribution is made.

Again, we show in figure 4 the results for the forward links only; the same increase in the estimation error with the order of the cumulants emerges also in this simulation run.

In order to test our technique in a more congested scenario, we increased the traffic load on each queue and repeated the simulation run; the results are shown in figure 5 and the estimation accuracy does not appear to be significantly affected by the increased link load.

In order to evaluate the heuristic described in section IV, we relied on model based Matlab simulations. The motivation for

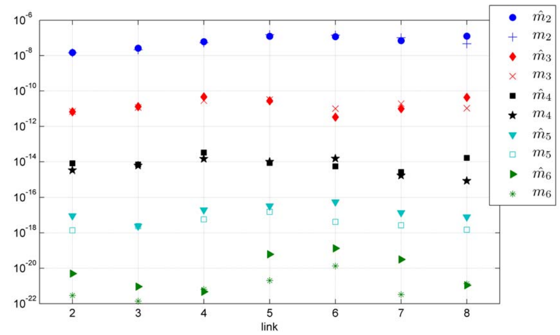

Figure 5. Estimated and actual cumulants of link delay distribution in a heavy-load scenario.

Table I
MEAN QUEUEING DELAY ESTIMATES IN A SCENARIO WITH EXPONENTIAL QUEUEING DELAY AND 4 CONGESTED LINKS

forw. est. μ	forw. actual μ	rev. est. μ	rev. actual μ
0.0036	0.0010	0	0.0020
0	0.0010	0.1009	0.1000
0	0.0010	0.0403	0.0400
0.0421	0.0400	0	0.0010
0	0.0010	0.0197	0.0200
0	0.0030	0.0048	0.0010
0	0.0010	0.0601	0.0600
0.0117	0.0100	0	0.0010

this choice is two-fold: first, model based simulation allows for a more strict control of the scenario, second, the calculation of the mean is not affected by the correlation among delays, which, on the contrary, influences the estimation of higher order cumulants. A first run of simulations has been performed by assuming again exponentially distributed delays and by hypothesizing the presence of only 4 congested links (the other links are assumed to be lightly loaded, i.e. associated with a mean queueing delay which is by at least an order of magnitude lower than that of the highly loaded links). The results are shown in table I and it clearly appears that, even if some errors affect the estimates associated with the less loaded links, the mean delay associated with congested links is generally well estimated.

In a second simulation run we alternated an equal number of heavily loaded links to almost idle links; the results are illustrated in table II and show our heuristic to correctly locate the congested links and to give a good approximation of their associated mean delay.

Again, the mean delay associated with the most congested links is generally well approximated, while that of the less congested ones is reduced to zero, in order to make the linear problem solvable.

CONCLUSIONS

In this paper, we propose an algorithm which allows to infer statistics concerning the queueing delay of each link of a network path (including both forward and reverse queues) based on measurements performed at the ends of the path only. In particular, our techniques is based on the estimation of the

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.

Table II
MEAN QUEUEING DELAY ESTIMATES IN A SCENARIO WITH EXPONENTIAL QUEUEING DELAY AND 8 CONGESTED LINKS

forw. est. μ	forw. actual μ	rev. est. μ	rev. actual μ
0	0.0010	0.0194	0.0200
0.0341	0.0300	0	0.0010
0	0.0010	0.0380	0.0400
0.0444	0.0400	0	0.0010
0	0.0010	0.0573	0.0600
0.0841	0.0800	0	0.0010
0	0.0010	0.0653	0.0600
0	0.0100	0	0.0010

cumulants associated with the delay distributions: we propose an unbiased estimator which allows to infer the cumulants with order higher than one as linear combinations of the cumulants of the distribution of the experimental data. As for the first order cumulants, we show that evaluating them based on end-to-end measurements is an undetermined problem. We then propose a heuristic method for estimating the mean value of the delay associated with the most congestioned links. Both the estimator and the heuristic are validated through several ns2 based and model based simulations and they prove to achieve a fairly good accuracy in the vast majority of cases.

REFERENCES

[1] ”The ns manual”, *http://www.isi.edu/nsnam/ns/doc/*

[2] Di Pietro, A. and Ficara, D. and Giordano, S. and Oppedisano, F. and Procissi, G ”End-to-end inference of link level queueing delay distribution and variance”, *International Symposium on Performance Evaluation of Computer and Telecommunication Systems, 2008. SPECTS 2008.*

[3] Meng-Fu Shih and Alfred O. Hero III ”Unicast-Based Inference of Network Link Delay Distributions with Finite Mixture Models”, *citeseer.ist.psu.edu/625167.html*

[4] M. Shih and A. Hero ”Unicast inference of network link delay distributions from edge measurements”, *Technical report, Comm. and Sig. Proc. Lab. (CSPL), Dept. EECS, University of Michigan, Ann Arbor, May 2001.*

[5] C. Rose and M. D. Smith ”k-Statistics: Unbiased Estimators of Cumulants.”, *Mathematical Statistics with Mathematica.*, New York: Springer-Verlag, pp. 256-259, 2002.

[6] F. LoPresti and N. G. Duffield and J. Horowitz and D. Towsley ”Multicast-Based Inference of Network-Internal Delay Distributions”, *University of Massachusetts 1999*

[7] Tsang, Yolanda and Yildiz, Mehmet and Barford, Paul and Nowak, Robert ”Network radar: tomography from round trip time measurements”, *IMC ’04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*

[8] N. G. Duffield and F. Lo Presti ”Network tomography from measured end–to–end delay covariance”, *IEEE/ACM Trans. Netw. 2004*

[9] N. G. Duffield and J. Horowitz and F. Lo Presti and D. Towsley ”Network Delay Tomography from End–to–End Unicast Measurements”, *Lecture Notes in Computer Science 2001*

[10] Van Jacobson ”Pathchar - a tool to infer characteristics of Internet paths.”, *Mathematical Sciences Research Institute(MSRI) 1997*

[11] G. Simon, P. Hága, G. Vattay, I. Csabai ”A Flexible Tomography Approach for Queueing Delay Distribution Inference”, *Proceedings of Internet Performance, Simulation, Monitoring and Measurement (IPS-MoMe 2005), p39-48, 14-15 March 2005, Warsaw, Poland (2005)*

[12] M. Coates and R. Nowak ”Network tomography for internal delay estimation”, *n Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Salt Lake City, UT, May 2001.*

[13] Yolanda Tsang, Mark Coates and Robert Nowak ”Nonparametric internet tomography”, *http://www.spin.rice.edu/PDF/icassp02.pdf*

[14] Arthur Dempster, Nan Laird, and Donald Rubin ”Maximum likelihood from incomplete data via the EM algorithm”, *Journal of the royal statistical society,1977*

[15] C. L. Lawson, R. J. Hanson ”Solving Least Square Problems”, *Prentice Hall, 1974*

[16] Gene H. Golub, Charles F. V ”Matrix Computations”, *Johns Hopkins Studies in Mathematical Sciences, 1996*

978-1-4244-4148-8/09/$25.00 ©2009
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2009 proceedings.