Neuro-Fuzzy Processing of Packet Dispersion Traces for Highly Variable Cross-Traffic Estimation

Marco A. Alzate[1,2], Néstor M. Peña[1], and Miguel A. Labrador[3]

[1] Universidad de los Andes, Bogotá, Colombia
[2]Universidad Distrital, Bogotá, Colombia
[3]University of South Florida, Tampa, FL, USA
{m-alzate,npena}@uniandes.edu.co, labrador@cse.usf.edu

Abstract. Cross-traffic data rate over the tight link of a path can be estimated using different active probing packet dispersion techniques. Many of these techniques send large amounts of probing traffic but use just a tiny fraction of the measurements to estimate the long-run cross-traffic average. In this paper, we are interested in short-term cross-traffic estimation using bandwidth efficient techniques when the cross-traffic exhibits high variability. High variability increases the cross-correlation coefficient between cross-traffic and dispersion measurements on a wide range of utilization factors and over a long range of measurement time scales. This correlation is exploited with an appropriate statistical inference procedure based on a simple heuristically modified neuro-fuzzy estimator that achieves high accuracy, low computational cost, and very low transmission overhead. The design process led to a very simple architecture, ensuring good generalization properties. Simulation experiments show that, if the variability comes from a complex correlation structure, a single estimator can be used over a long range of utilization factors and measurement periods with no additional training.

Keywords: Traffic estimation; Packet pair dispersion; Neuro-fuzzy systems.

1 Introduction

Several network parameters and traffic conditions can be inferred from packet dispersion measurements, when a sender transmits probing packets of given length at given instants of time, and a receiver collects them taking note of their inter-arrival times [1]. For example, if the tight link is 100% busy between a pair of probing packets, the correlation coefficient between the dispersion measurements and the tight-link cross-traffic will be 1, i.e., the dispersion measurement reveals the average cross-traffic rate over that link [2]. Otherwise, this correlation will be less than one, increasing directly with the utilization factor and inversely with the probing packets inter-departure times.

Most available bandwidth estimation techniques send large amounts of probing traffic (overhead) in order to select the tiny fraction of measurements that satisfies the high correlation condition [3][4][5][6]. However, several simulation experiments with different synthetic traces exhibiting different degrees of variability (not shown here) reveal that this correlation can still be high over a wide range of link utilizations and over a long range of measurement time scales if the traffic's coefficient of variation is high and the traffic exhibits long range dependence. In this work, we consider a computational intelligence approach to estimate the competing traffic rate in the tight link that, instead of ignoring those measurements during which the tight-link becomes idle, it exploits the correlation that still exists between those dispersion measurements and the bursty cross-traffic under high variability conditions.

2 Neuro-Fuzzy Cross-Traffic Estimator

Consider two probing packets of length L bits, sent T seconds apart over a FIFO link of capacity C bps, so that the queue does not empty between the departure of the first packet and the arrival of the second packet. The dispersion D will be L/C plus the time taken to transmit the cross-traffic that arrived during T. Consequently, we can estimate the average cross-traffic rate during that period of length T as $X = (D \cdot C - L)/T$. If there is no cross-traffic or the cross-traffic is small enough to be completely transmitted between probe packet arrivals, both probe packets will find an empty queue and the dispersion will be $D = T$. In any other case, when one or both probe packets find a non-empty queue but there are empty periods between the departure of the first packet and the arrival of the second packet, the dispersion is a random variable more or less correlated with the cross-traffic process, depending on the fraction of time the link was idle.

Figure 1 shows the results of a simple simulation experiment where we used the estimator above on the Bellcore traffic trace BCpAug89 [7] when it shares a T1 link with probe packets sent every second. The upper plot shows the true and estimated traffic during a small period, while the lower plot shows the corresponding buffer length. The estimator is exact when the link is 100% busy (900 to 940 seconds) and poor when the link is almost completely idle (990 to 1030 seconds). An intermediate performance corresponds to a not too busy link (1065 to 1085 seconds).

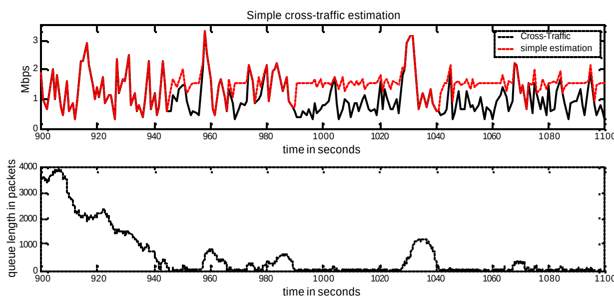


Fig. 1. Performance of the simple estimator.

Let m_D and $s_D{}^2$ be the mean and the variance of the previous k dispersion measures. By observing Figure 1, we can establish plausible rules, such as "If m_D is far from T, the simple estimation is exact", "If m_D is close to T and $s_D{}^2$ is small, the simple estimation is a poor one", and "If m_D is close to T and $s_D{}^2$ is large, the simple estimation is fair". This reasoning calls for a fuzzy approach to our estimation problem, for which we use a neuro-fuzzy system that allows us to construct an estimator suited to our particular training data (D_n, X_n), where D_n is the n^{th} dispersion measurement and X_n is the corresponding cross-traffic rate. We decided to use D_n, D_{n-1}, and the sample mean and variance of the last 12 measurements as input variables, because they form a small set that have almost as much mutual information with X_n as the joint 12 dispersion measurements together. Next, traffic is centered and normalized with respect to the known capacity, C, while the dispersion measurements are centered and normalized with respect to T,

$$x_n = \frac{X_n - C}{C}, \qquad d_n = \frac{D_n - T}{T} \tag{1}$$

So that the selected inputs to the traffic estimation system become

$$q_1(n) = d_n \qquad q_2(n) = d_{n-1} \qquad q_3(n) = \frac{1}{12}\sum_{k=0}^{11} d_{n-k} \qquad q_4(n) = \frac{1}{11}\sum_{k=0}^{11}\left(d_{n-k} - q_3(n)\right)^2 \tag{2}$$

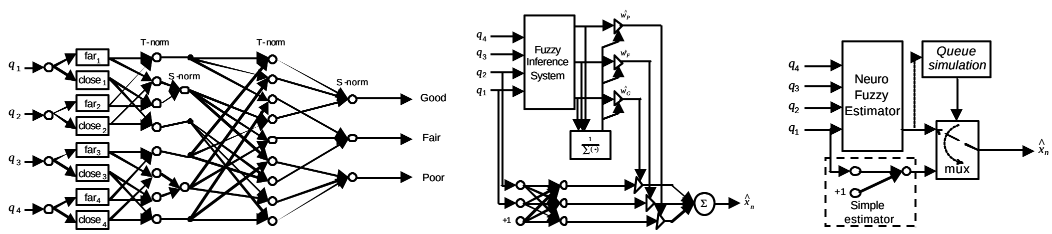
and the estimator $X_n = (D_n \cdot C - L)/T$ becomes the following simple estimator (*SE*)

$$\hat{x}_n = d_n - L/(C \cdot T) \tag{3}$$

Fitting the histograms of each input variable conditioned on an "exact" or "poor" performance of the *SE*, we define fuzzy sets "Far from zero" and "Close to zero" through the following membership functions,

$$\begin{aligned} &m_{CLOSE_i}(q_i) = \exp(-l_i\,|q_i|), \quad m_{FAR_i}(q_i) = 1 - m_{CLOSE_i}(q_i) \quad \text{for } i\text{=1,2,3} \quad \text{with } l_1 = l_2 \\ &m_{CLOSE_4}(q_4) = \exp(-l_4 q_4), \quad m_{FAR_4}(q_4) = 1 - m_{CLOSE_4}(q_4) \end{aligned} \tag{4}$$

Now we use three rules for deciding whether *SE* is 'good', 'fair' or 'poor' and, for each case, we compute an affine transformation of q_1 and q_2. This way, we obtain only three non-linear parameters and nine linear ones, increasing regularity, simplifying the training process, and reducing the computational cost. We also use a simple simulation formula to estimate the queue length so that, if it is bigger than a given threshold (*thr*, an additional non-linear parameter), the *SE* is selected. The heuristically modified neuro-fuzzy estimator (*HNFE*) is shown in Figure 2.



(a) (b) (c)

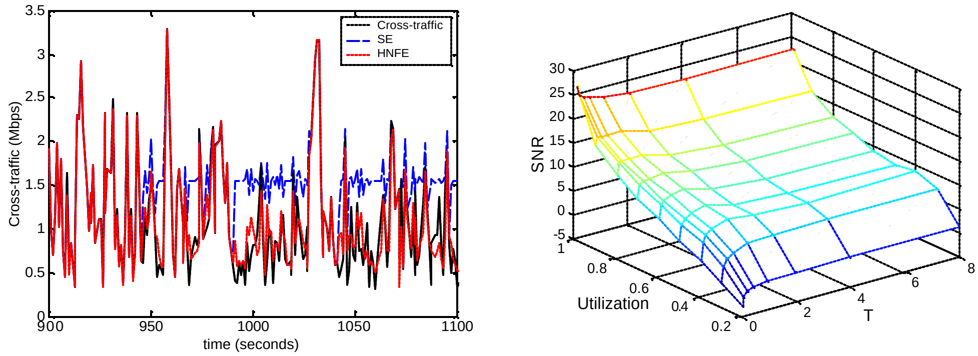
Fig. 2. The fuzzy inference system for classifying the quality of *SE*, (a), is used to combine the outputs of three appropriate affine estimators, (b). The queue heuristic is used to recover the simple estimator (c).

In a preliminary step for training, we initialize l_1, l_3, and l_4 by fitting the conditional histograms. The first step is the computation of the optimal linear parameters by a least square procedure. The second step is the computation of the optimal exponents through a quasi-Newton line search algorithm. These two steps are iterated until convergence. Finally, we look for the optimal *thr* through bracketing. The algorithm is so efficient that on the first evaluation of step 1, we already increased the signal-to-noise ratio (where the traffic trace is the signal and the estimation error is the noise) from 2 dB for *SE* to 9.3 dB for *HNFE*. After four iterations, we achieved 14.2 dB, and when we added the queue heuristic, we obtained a final performance of 14.9 dB on the training data. The whole procedure took a few seconds on a typical PC, for a one-hour traffic trace.

3 Numerical Results and Conclusions

Figure 3(a) reproduces Figure 1 with the new *HNFE* estimator. To check for generalization properties, we used the same *HNFE* estimator, without additional training, on a different traffic trace (a 768 kbps MPEG4 version of "Jurassic Park" [8]) under different link utilization factors, *r*, and different measurement periods, *T*. The estimation SNR is shown in Figure 3(b). The system exhibits a remarkable invariance with time scale and results very useful in a long range of traffic intensities, without any modification to the trained *HNFE*.

Concluding, in this paper we show that the high variability of modern networks traffic can be conveniently exploited for better instantaneous estimations within the range of time scales at which the high variability is exhibited. We devised a heuristically modified neuro-fuzzy cross-traffic estimator that is highly accurate, even for low long-run traffic intensities, as long as the variability allows for the presence of measurement periods with high link occupancy. This accuracy is achieved with very low computational complexity and at a minimal transmission overhead.



(a) (b)

Fig. 3. Performance of the *HNFE* on the training data (a). SNR under different network conditions for a video cross-traffic trace (b).

References

1. R.S. Prasad, M. Murray, C. Dovrolis, and K.C. Claffy, "Bandwidth Estimation: Metrics, Measurement Techniques, and Tools," IEEE Network Magazine, Vol. 17, No. 6, pp. 27--35, Nov/Dec. 2003.
2. N. Hu and P. Steenkiste, “Evaluation and Characterization of Available Bandwidth Probing Techniques,” IEEE JSAC, Vol. 21, No. 6, pp. 879-894, Aug., 2003.
3. M. Jain and C. Dovrolis, “End-to-End Available Bandwidth Measure Methodology, Dynamics and Relation with TCP throughput,” IEEE/ACM Transactions on Networking, Vol. 11, No. 4, August 2003, pp. 537-549.
4. V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil and L. Cottrell, “PathChirp: Efficient Available Bandwidth Estimation for Network Paths,” Proceedings of Passive and Active Measurements (PAM) Workshop, La Jolla, CA, USA, Apr. 2003.
5. J. Strauss, D. Katabi, F. Kaashoek, and B. Prabhakar, “Spruce: A Lightweight End-to-End Tool for Measuring Available Bandwidth,” Proceedings of Internet Measurement Conference (IMC) 2003, Miami, Florida, October 2003.
6. R. Ribeiro, M. Coates, R. Riedi, S. Sarvotham, B. Hendricks, and R. Baraniuk, “MultiFractal Cross-Traffic Estimation,” Proceedings of ITC Specialist Seminar on IP Traffic Measurement, Monterey California, September 18-20 2000
7. Lawrence Berkeley National Laboratory. The Internet Traffic Archives, BC – Ethernet traces of LAN and WAN traffic, http://ita.ee.lbl.gov/html/contrib/BC.html
8. Video Traces Research Group, Arizona State University, http://trace.eas.asu.edu/TRACE/pics/FrameTrace/mp4/Verbose_Jurassic.dat.