A Bayesian Approach for TCP to Distinguish Congestion from Wireless Losses ⋆

Dhiman Barman and Ibrahim Matta

Department of Computer Science
Boston University
Boston, MA 02215, USA
{dhiman,matta}@cs.bu.edu

Technical Report BUCS-2003-030

Abstract. The Transmission Control Protocol (TCP) has been the protocol of choice for many Internet applications requiring reliable connections. The design of TCP has been challenged by the extension of connections over wireless links. In this paper, we investigate a Bayesian approach to infer at the source host the reason of a packet loss, whether congestion or wireless transmission error. Our approach is "mostly" end-to-end since it requires only one *long-term average* quantity (namely, long-term average packet loss probability over the wireless segment) that may be best obtained with help from the network (e.g. wireless access agent). Specifically, we use Maximum Likelihood Ratio tests to evaluate TCP as a classifier of the type of packet loss. We study the effectiveness of *short-term* classification of packet errors (congestion vs. wireless), given stationary prior error probabilities and distributions of packet delays conditioned on the type of packet loss (measured over a longer time scale). Using our Bayesian-based approach and extensive simulations, we demonstrate that an efficient online error classifier can be built as long as congestion-induced losses and losses due to wireless transmission errors produce sufficiently different statistics. We introduce a simple queueing model to underline the conditional delay distributions arising from different kinds of packet losses over a heterogeneous wired/wireless path. To infer conditional delay distributions, we consider Hidden Markov Model (HMM) which explicitly considers discretized delay values observed by TCP as part of its state definition, in addition to an HMM which does not as in [9]. We demonstrate how estimation accuracy is influenced by different proportions of congestion versus wireless losses and penalties on incorrect classification.

1 Introduction

Many studies have analyzed the performance of transport protocols, notably TCP [7]. TCP carries most of the traffic—around 90% of the bytes—in the Internet [2]. TCP has been designed to do congestion control to achieve efficient and fair allocation of resources within the network.

In a wired network, congested links cause packets to get lost when the bottleneck buffer overflows. If a TCP connection traverses a wireless link, for example a WLAN network, packets may be corrupted and get lost due to fading or shadowing. Such wireless losses are *not* an indication of resource scarcity in the routers and it is intuitive that an informed transport protocol would treat such packet losses differently. The performance of an ideal informed TCP has been shown in [5] in which the TCP sender does not back off on wireless losses. But for an end-to-end protocol, inferring the nature of loss without any aid from the network is challenging. Nevertheless, many proposals [9, 6] attempted to infer (implicitly or explicitly) the reason of a packet loss, in an end-to-end way, by analyzing measured delays, throughput or other metric.

Our approach is explicit and "mostly" end-to-end. We say "mostly" since our technique requires only one *long-term average* quantity (namely, long-term average packet loss probability over the wireless segment) that may be best obtained with help from the network (e.g. wireless access agent). We elucidate

⋆ This work was supported in part by NSF grants ANI-0095988, EIA-0202067, and ITR ANI-0205294. An abbreviated version of this paper appears in *WiOpt'04: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, March 2004, University of Cambridge, UK, under the title "Model-based Loss Inference by TCP over Heterogeneous Networks."

the difference in the output (measured) statistics under different type of losses (congestion vs. wireless), and exploit those using signal estimation techniques such as Maximum Likelihood Estimation. It is to be disclaimed that we are *not* overlooking the better performance that may result from "heavier" infrastructure support, e.g. XCP [8] and Snoop [4]. Such infrastructures have their own cost of deployment and may not be effective, for example with IPsec [11].

In distinguishing the cause of packet loss, we exploit the temporal correlation between losses and the measured end-to-end metrics. Congestion-induced losses are associated with (close to) full buffer size at the bottleneck, whereas wireless losses often sample any queue size and associated delays. This leads to *distinguishable* distributions of the measured samples of network response at the times of different type of loss.

Figure 1 shows our model where measured samples are noisy observations in the vicinity of the losses. Network conditions result in an output (e.g. packet loss due to congestion or wireless error), which we

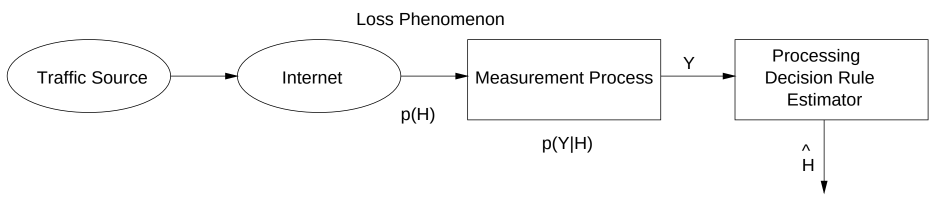


Figure 1. Elements of the Detection Problem

classify by a *hypothesis* and denote by H. This outcome generated by the network state is carried by packet samples after a certain time lag, thus the samples are probabilistically affected and serve as observation samples Y. Based on the observation samples, we intend to design a rule to decide what the cause of the loss is (i.e., whether the hypothesis that the loss is due to congestion or wireless error holds). Such a model assumes knowledge of the apriori (actual) probability of a hypothesis, denoted by $P(H)$, and the probability distribution of the observed metric Y conditioned on the hypothesis being true, denoted by $P(Y|H)$. We later discuss how this prior knowledge can be obtained.

Our goal is to obtain the best possible estimate, $\hat{H}$ (representing the classification of a packet loss as congestion-induced or due to wireless error), that minimizes the average penalty of misclassifying the type of loss. This would give us a handle on the theoretical limits and gains of end-to-end packet error classification.

To that end, we use Bayes Decision Rules and Maximum Likelihood Ratio Tests. The penalty function should measure the dissatisfaction of the application of its performance. For example, if the network is congested and the protocol misclassifies a packet loss, i.e. the loss is not attributed to congestion rather to a wireless loss, this congestion loss misclassification may incur more cost than an otherwise wireless loss misclassification. This could be due to increased congestion as the source did not react appropriately (backed off) in response to actual congestion. Therefore, it makes sense to map any observation to a hypothesis which will reduce the cost of classification error. Using Bayes Rule, we have

$$P(H|Y) = \frac{P(Y|H)P(H)}{P(Y)} \tag{1}$$

From Equation (1), it follows that if we know the prior probability of Y under some hypothesis H, the prior probabilities H and the unconditional probability of Y, we can derive the probability of a hypothesis from Y. In practice, the priors – $P(Y|H)$ and $P(H)$ – will be measured/estimated over time scale that is longer than that of the short-term goal of packet error classification.

Paper Contributions: We use Maximum Likelihood Ratio tests to evaluate TCP as a detector/estimator of the type of packet loss. We study the effectiveness of short-term classification of packet errors (congestion vs. wireless) given stationary prior error probabilities and conditional delay distributions measured over a longer time scale. Using our model-based approach and extensive simulations, we demonstrate that an efficient online detector can be built as long as congestion-induced losses and losses due to wireless transmission errors produce sufficiently different statistics. We introduce a simple queueing model to underline the conditional delay distributions arising from different kinds of packet losses over a heterogeneous wired/wireless path. To infer conditional delay distributions, we consider Hidden Markov

Model (HMM) which explicitly considers discretized delay values observed by TCP as part of its state definition, in addition to an HMM which does not as in [9]. We train the HMM using either all measured delays or only those delays measured by loss pairs—a loss pair is a pair of back-to-back packets where one packet is lost and the other one is used to infer the state of the path around the time of loss [9]. We demonstrate how estimation accuracy is influenced by different proportions of congestion versus wireless losses and penalties on incorrect estimation.

Paper Outline: Section 2 introduces Bayesian hypothesis testing for making a binary decision. Section 3 presents a queueing model and simple analysis of delay distributions conditioned on the type of loss, and discusses an HMM-based scheme to infer conditional delay distributions. Section 4 instantiates Bayesian binary testing assuming Gaussian conditional delay distributions. The accuracy of classification is defined in Section 5, and Section 6 presents validation results using ns-2 simulation [3]. Section 7 concludes the paper with future work.

2 Bayesian Binary Hypothesis Testing

In this section, we use Bayesian binary hypothesis testing to infer the reason of packet loss. We consider the simplest classification—a packet loss is either due to congestion (i.e. buffer overflow) or due to wireless (i.e. transmission error). So we have two possible network states, which we label through hypotheses $\mathcal{C}$, corresponding to "congestion loss hypothesis", and $\mathcal{W}$, corresponding to "wireless loss hypothesis." In our approach we have three models: (i) a model of the network state, (ii) a model of the observations, and (iii) decision rules. The model of the network state is captured by the prior probabilities, $P(C)$ and $P(W)$—the actual probabilities of a congestion-induced and wireless loss, respectively. The observation model captures the relationship between the observed quantity y and $P(C)$ or $P(W)$ (measured over a long time scale) by the conditional densities $P(y|C)$ or $P(y|W)$. Our decision rule $D(y)$ is obtained by minimizing the average cost ("Bayes risk").

Let R_{wc} denote the cost of deciding that $D(y) = \mathcal{W}$ when the actual cause of loss is congestion (i.e., misclassifying congestion loss). Similarly, we denote by R_{cw} the cost of deciding that $D(y) = \mathcal{C}$ when the actual cause of loss is wireless (i.e., misclassifying wireless loss). In our formulation, we assume that the penalty of misclassification of losses is constant. Then the Bayes risk of the decision rule is given by:

$$\begin{aligned} E[R_{D(y)}] &= R_{cw}P(D(y) = \mathcal{C}, W) + R_{wc}P(D(y) = \mathcal{W}, C) \\ &= E[E[R_{D(y)}|y]] = \int E[R_{D(y)}|y]P(y)dy \end{aligned} \tag{2}$$

From Equation (2), we can minimize the penalty of misclassification by minimizing $E[R_{D(y)}|y]$ for each value of the observed sample, y. Thus, the optimal decision is to choose the hypothesis that yields the smallest value of the conditional penalty cost $E[R_{D(y)}|y]$ for a given value of y. The conditional expected penalty is given by:

$$E[R_{D(y)}|y] = R_{cw}P(D(y) = \mathcal{C}, W|y) + R_{wc}P(D(y) = \mathcal{W}, C|y) \tag{3}$$

For a given observation value y, the expected value of the conditional penalty if we choose to assign the observation to $\mathcal{W}$ or $\mathcal{C}$ is given by:

$$\text{If } D(y) = \mathcal{W} : E[R_{D(y)}|y] = R_{wc}P(C|y) \tag{4}$$

$$\text{If } D(y) = \mathcal{C} : E[R_{D(y)}|y] = R_{cw}P(W|y) \tag{5}$$

Given the above conditions, the optimal decision is one that results in the smaller of the two conditional costs. Using Bayes rule and reorganizing Equations (4) and (5), we have:

$$\begin{aligned} P(C)R_{wc}P(y|C) &\underset{\mathcal{W}}{\overset{\mathcal{C}}{\gtrless}} P(W)R_{cw}P(y|W) \\ \mathcal{L}(y) = \left[\frac{P(y|C)}{P(y|W)}\right] &\underset{\mathcal{W}}{\overset{\mathcal{C}}{\gtrless}} \frac{R_{cw}P(W)}{R_{wc}P(C)} \equiv \Gamma \end{aligned} \tag{6}$$

where $\underset{\mathcal{W}}{\overset{\mathcal{C}}{\gtrless}}$ denotes choosing $\mathcal{C}$ (i.e., classifying a packet loss due to congestion) if the inequality is $>$, and choosing $\mathcal{W}$ (i.e., classifying a packet loss due to wireless error) if the inequality is $<$.

In the network, the packet samples carry the network state information to the receivers or senders delayed by propagation time. Moreover, those samples which make their way through the network are only discrete samples of the network state. But due to temporal and spatial locality, we assume that samples received immediately around the time of loss have most energy and information of the state. This is also an objective of our expedition to know how effective are these samples.

The test (i.e., sufficient statistics) in Equation (6) assumes the knowledge of priors, $P(C)$ and $P(W)$ and conditional delay distributions, $P(y|C)$ and $P(y|W)$ which are in general very difficult to obtain. If we can know these unknowns or at least estimate them accurately, we can apply the Bayesian Likelihood tests in Equation (6). Our next section is devoted to describing how we can obtain conditional delay distributions for a simple queueing system, or in practice how we can estimate them using Hidden Markov Models. We can estimate long-term average values of the priors, $P(C)$ and $P(W)$, using different existing tools [10, 12] with router support. For example, the TCP receiver can measure the long-term wireless link error rate with support from the base station and inform the TCP sender. Given $P(W)$ and the end-to-end loss probability measured by the sender, the sender can obtain $P(C)$. Note that in the subsequent queueing model, we easily estimate the prior $P(C)$, which is simply the packet drop probability when the buffer overflows.

3 Estimation of Conditional Delay Distributions

3.1 Queueing Model

Assume the TCP receiver measures one-way delay of the packets which are in the same transmission window as the packet which is dropped. We derive the conditional probabilities of delays based on the assumption that there is one bottleneck (M/M/1/B) queue, where a packet is dropped if the packet arrives to a full queue. We model that packets are only delayed but not dropped in the upstream queues using an (M/M/1/∞) queue. Packets are also dropped due to wireless error on links which may be located before or after. Here we assume wireless links appear after the wired bottleneck, although the location of wireless links before the wired bottleneck will not change the analysis. If delay statistics are collected by the packets immediately preceding the dropped packet, we analyze the conditional delay probabilities for such packets.

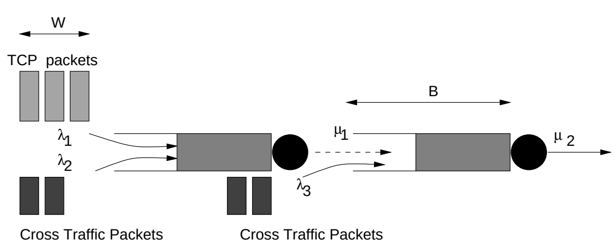


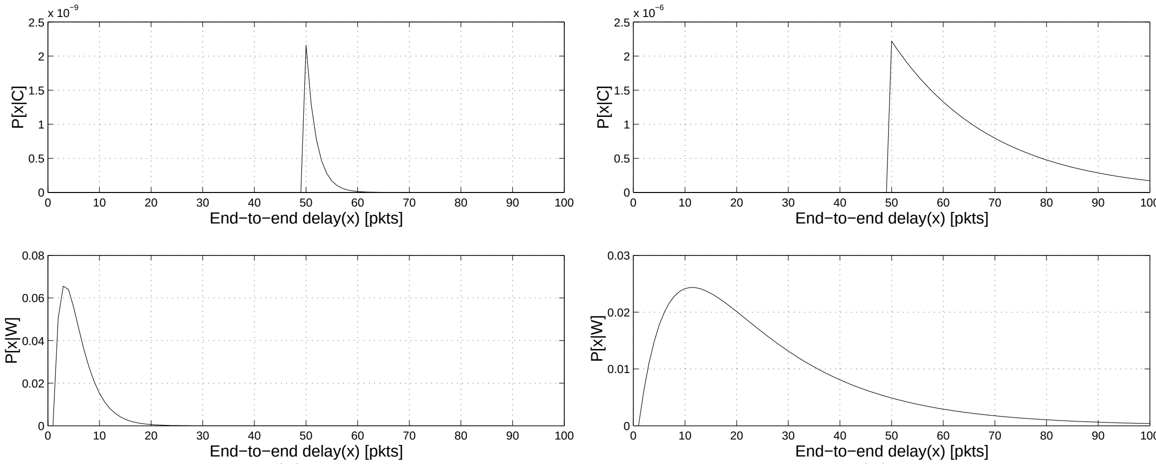
Figure 2. Arrival of TCP packets at Bottleneck

Consider Figure 2 that shows TCP packets as bursty Poisson arrivals with average rate λ_1 to an (M/M/1/∞) queue, where W is assumed to be the burst (window) size. The queue is shared by cross traffic packets which arrive according to a Poisson distribution at average rate λ_2. Thus, the load on the queue is given by $\rho_1 = \frac{\lambda_1 W + \lambda_2}{\mu_1}$. For simplicity we assume a fixed burst size—in general burst sizes are correlated in TCP—and we also assume there is no packet loss in this queue.

Next we consider a downstream queue (M/M/1/B). We assume packets after leaving the first non-bottleneck queue still remain bursty poisson arrivals. TCP packets arrive in bursts with average rate λ_1. Cross traffic packets arrive with average rate λ_3. The packet loss probability of the TCP flow is given by:

$$P(C) = \pi(B) + \pi(B-1)\tfrac{W-1}{W} + \cdots + \pi(B-W+1)\tfrac{1}{W} \tag{7}$$

$$= \frac{(1-\rho_2)\rho_2^B}{1-\rho_2^{B+1}} \left[\frac{1-(\frac{1}{\rho_2})^W}{1-\frac{1}{\rho_2}} - \frac{\frac{1}{W} + \frac{W-1}{W\rho_2^W} - \frac{1}{\rho_2^{W-1}}}{\rho_2(1-\frac{1}{\rho_2})^2} \right] \tag{8}$$



(a) (b)

Figure 3. Conditional Delay Distributions for (a) $B = 50, \rho_1 = 0.6, \rho_2 = 0.7$; (b) $B = 50, \rho_1 = 0.95, \rho_2 = 0.85$

where π is the stationary distribution of the total number of packets in the bottleneck queue. The load on the bottleneck queue is given by $\rho_2 = \frac{\lambda_1 W + \lambda_3}{\mu_2}$. We consider that there exists a wireless link further downstream in which packets are dropped with average loss rate of $P(W)$.

First, we derive the delay distribution of the packet preceding the dropped one conditioned on congestion loss. Note that the minimum delay of such a packet is at least $B+1$ (i.e., $B-1$ of the bottleneck queue size and 2 due to packet being in service twice) because the dropped packet arrives to the bottleneck queue when its size is B. Also, the same packet experiences some delay in the upstream queue. If $x \geq B+1$, where x is end-to-end delay in terms of packets, then $P(X = x|C) = (1-\rho_1)\rho_1^{x-B}[\pi(B-1)B + \pi(B-2)B + \cdots + \pi(B-W+1)B]$. Thus, we have:

$$P(X = x|C) = \begin{cases} (1-\rho_1)\rho_1^{x-B}\frac{B(1-\rho_2)^{W+1}\rho_2^{B-1}}{1-\rho_2^{B+1}} & \text{if } \ x \geq B+1 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

where X is a random variable representing end-to-end delay.

Now, we consider the delay distribution of the packet preceding the dropped one on the wireless link. Note that both the dropped packet and its previous packet make it through the bottleneck queue. Thus, the previous packet experiences an end-to-end delay which is cumulative of the queueing delays from both the queues. If $x \geq B+1$, then we have $P(X = x|W) = \sum_{m=1}^{B}(1-\rho_1)\rho_1^{x-m}\frac{(1-\rho_2)\rho_2^m}{1-\rho_2^{B+1}}$, and if $x < B+1$ then $P(X = x|W) = \sum_{m=1}^{x}(1-\rho_1)\rho_1^{x-m}\frac{(1-\rho_2)\rho_2^m}{1-\rho_2^{B+1}}$. Thus, we have,

$$P(X = x|W) = \begin{cases} \frac{(1-\rho_1)(1-\rho_2)}{1-\rho_2^{B+1}}\rho_1^{x-1}\rho_2\frac{1-(\frac{\rho_2}{\rho_1})^x}{1-\frac{\rho_2}{\rho_1}} & \text{if } \ x < B+1 \\ \frac{(1-\rho_1)(1-\rho_2)}{1-\rho_2^{B+1}}\rho_1^{x-1}\rho_2\frac{1-(\frac{\rho_2}{\rho_1})^B}{1-\frac{\rho_2}{\rho_1}} & \text{otherwise} \end{cases} \tag{10}$$

We plot the delay distribution of the packet preceding the lost one conditioned on congestion loss (Equation 9) and wireless loss (Equation 10) in Figure 3.

The main emphasis is to portray the fact that the conditional delay distributions are distinguishable. We do notice that due to the simplistic assumptions of the queueing model, the analytical distributions may not be identical to empirical distributions but our intent is to highlight the difference between the two conditional distributions and gain insight of the loss phenomenon through an intuitive model. All the previous works have tried to identify or distinguish the nature of loss without providing any detailed queueing model but using some measurement heuristics. It is a future research direction to develop a complete queueing model so that TCP can be evaluated more accurately and also desired control can be incorporated into the TCP error control mechanism. Note that as the system becomes more overloaded, the two conditional distributions tend to overlap, which increases the likelihood of error in classification. This fact was also emphasized experimentally in [9].

3.2 An EM Algorithm to infer the Conditional Delay Distributions

We now describe the inference for $P(y|C)$ and $P(y|W)$, using a Hidden Markov Model which explicitly considers discretized delay values as part of its state definition (similar to the one developed in [15]). We refer to a discretized RTT delay as delay symbol. Suppose there are M delay symbols and N hidden states. Each state of the model Z_t contains two components: the hidden state $X_t \in \{1, 2, \ldots, N\}$ and the delay symbols $Y_t \in \{1, 2, \ldots, M\}$. That is $Z_t = (X_t, Y_t)$.

Let π denote the initial distribution of the states. Let P denote the probability transition matrix. An element in the transition matrix P is denoted as $P_{(i,j)(k,l)}$, which represents the transition probability from state (i, j) to state (k, l). Let y_t be the observation value of Y_t. If at time t, the observation is a loss, we regard it as a delay with a missing value and use $y_t = *$ to denote it. A loss observation has a certain probability of having a delay symbol of $j, 1 \leq j \leq M$. Let $s(j)$ be the conditional probability that an observation is a loss given that its delay symbol is j (if it were not lost). That is, $s(j) = P(y_t = *|y_t = j)$. Let $s(i, j)$ be the conditional probability that an observation is a loss given that its delay symbol is j and the hidden state is i. That is, $s(i, j) = P(y_t = *|Z_t = (i, j))$. Let $\lambda = (P, \pi, s)$ denote the complete parameter set of the model. An Expectation Maximization (EM) algorithm is an iterative procedure to infer λ from a sequence of T observations. Given λ, we can determine $P(Z_t = (i, j)|y_t = *)$. Thus, given that hidden state i corresponds to a congestion state, we associate a loss observation to congestion. Otherwise, we associate a loss observation to wireless error. Throughout this paper, we classify the state corresponding to higher mean delay as "congestion" state and the other as "wireless error" state. The EM algorithm ends when a certain convergence threshold is reached. Unless otherwise specified, we show results for $N = 2$ and $M = 10$.

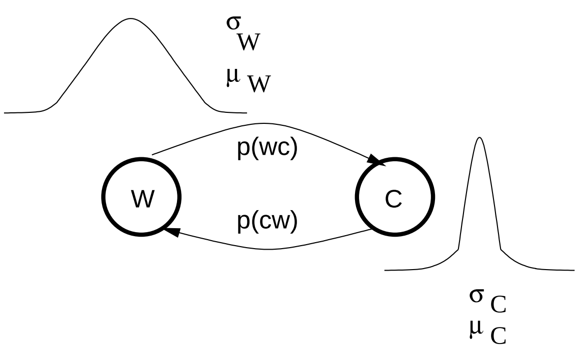


Figure 4. Hidden Markov Model

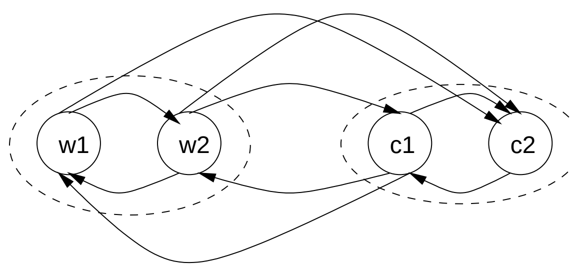


Figure 5. Hidden Markov Model with hidden delay dimension

In order to estimate the conditional delay distributions, we obtain the empirical delay distributions estimated by the HMMs illustrated in Figures 4 and 5. Figure 4 is a regular HMM which is trained using loss pairs. From such a trained HMM, we obtain conditional delay distributions assuming that the state corresponding to higher mean delay represents congestion while the other state represents wireless error. We denote the mean and variance of the delay distributions at the congestion state by μ_c and σ_c, respectively, and similarly at the wireless state, by μ_w and σ_w. We denote the transition probability from the wireless state to the congestion state by P_{wc}, and the reverse transition probability by P_{cw}.

In Figure 5, we illustrate the HMM (with hidden dimension) through an instance in which $N = 2$ and $M = 2$. Therefore, wireless states coupled with discretized delay symbols are represented by $w1$ and $w2$. Similarly, congestion states are represented by $c1$ and $c2$. The delay symbols represent part of each state and thus this HMM model can capture temporal correlation well and such a model can be trained using any delay sample and not just loss pairs.

4 Gaussian Model

We present a simple case in which we approximate conditional delay distributions by Gaussian distributions. On average, all packet samples before congestion losses should see an almost full bottleneck queue size.[1] Denote the corresponding average delay by μ_c. Since cross traffic and the measurement process

[1] How close to a full buffer depends on the behavior of the cross-traffic.

introduce noise, we see a perturbed sample value. We assume that noise is white Gaussian with zero mean and variance σ_c^2.

Prior to wireless losses, the delay samples see a lower average delay. Denote the corresponding average delay by μ_w. Again, we assume that noise is white Gaussian with zero mean and variance σ_w^2.We can now pose the detection or classification of congestion versus wireless loss as a scalar Gaussian detection problem:

$$\mathcal{C} : y = \mu_c + N(0, \sigma_c^2) \quad \mathcal{W} : y = \mu_w + N(0, \sigma_w^2)$$

Substituting in Equation (6), we get:

$$\mathcal{L}(y) = \left[\frac{\left(\frac{1}{\sqrt{2\pi\sigma_c^2}}\right) e^{-\frac{(y-\mu_c)^2}{2\sigma_c^2}}}{\left(\frac{1}{\sqrt{2\pi\sigma_w^2}}\right) e^{-\frac{(y-\mu_w)^2}{2\sigma_w^2}}} \right] \underset{\mathcal{W}}{\overset{\mathcal{C}}{\gtrless}} \frac{R_{cw}P(W)}{R_{wc}P(C)} \tag{11}$$

Taking natural logarithm on both sides and rearranging the terms, we have:

$$-\frac{(y-\mu_c)^2}{2\sigma_c^2} + \frac{(y-\mu_w)^2}{2\sigma_w^2} \underset{\mathcal{W}}{\overset{\mathcal{C}}{\gtrless}} \ln\left(\frac{\sigma_c \Gamma}{\sigma_w}\right) \tag{12}$$

where $\Gamma = \frac{R_{cw}P(W)}{R_{wc}P(C)}$. Figure 6 illustrates the two possible loss scenarios. Γ determines the degree of correct classification (or misclassification). The area denoted by P_D represents the correct classification of congestion (henceforth denoted by $P(\mathcal{C}|C)$), whereas the area denoted by P_F represents the misclassification of wireless loss as congestion-induced (henceforth denoted by $P(\mathcal{C}|W)$). The value of Γ depends on the penalties of misclassification as well as the ratio of wireless to congestion loss probabilities. Note that in practice, the two penalties of misclassifying loss type, R_{cw} and R_{wc}, are not necessarily equal and greatly depend on the source behavior, in our case, TCP. Furthermore, the degree of wireless losses, $P(W)$, affects the sending rate of TCP, which in turn determines the degree of congestion losses, $P(C)$. In this paper, we vary the value of Γ so we quantify the potential gains and limits of end-to-end error classification.

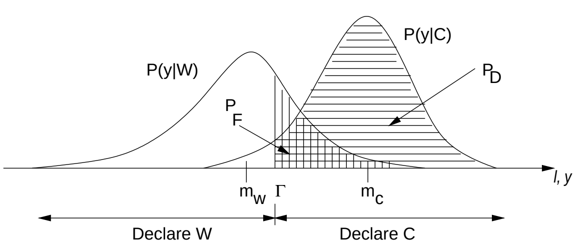


Figure 6. Scalar Gaussian case for binary hypothesis testing for distinguishing between congestion loss and wireless loss

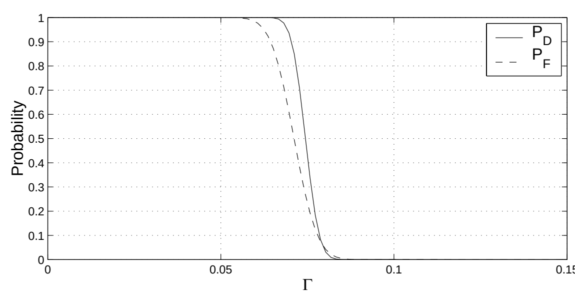


Figure 7. P_D and P_F for the setup in Figure 8

5 Accuracy of Classification

To evaluate the performance of a decision rule, we consider the probability of misclassification error, Pr[Error] expressed using Bayes rule as follows:

$$\Pr[\text{Error}] = P(\mathcal{W}|C)P(C) + P(\mathcal{C}|W)P(W) \tag{13}$$

Note that $P(\mathcal{C}|C) + P(\mathcal{W}|C) = 1$, and that $P(\mathcal{C}|W) + P(\mathcal{W}|W) = 1$. Thus, we can determine the performance of a decision by calculating the probabilities of correct loss-type classification, $P(\mathcal{C}|C)$ and $P(\mathcal{W}|W)$, which should be maximized. We know that Maximum Likelihood Tests are optimal and thus we focus on knowing the values of $P(\mathcal{C}|C)$ and $P(\mathcal{W}|W)$ for every possible value of the threshold, $\Gamma = \frac{R_{cw}P(W)}{R_{wc}P(C)}$ (cf. Equation (6)).

To that end, from Equation (6), expressing a general decision rule test as $\mathcal{L}(y) \underset{\mathcal{W}}{\overset{\mathcal{C}}{\gtrless}} \Gamma$ where $\mathcal{L}(y)$ is a random variable, we have:

$$P(\mathcal{C}|C) = \int_{\{y|\mathcal{C}\}} P(y|C)dy = \int_{\mathcal{L}>\Gamma} P(\mathcal{L}|C)d\mathcal{L}$$

$$P(\mathcal{W}|W) = 1 - P(\mathcal{C}|W) = 1 - \int_{\{y|\mathcal{C}\}} P(y|W)dy = 1 - \int_{\mathcal{L}>\Gamma} P(\mathcal{L}|W)d\mathcal{L}$$

In our scalar Gaussian detection problem (cf. Equation (11)), we have $P(\mathcal{C}|C) = Q\left(\frac{\Gamma-\mu_c}{\sigma_c}\right)$ and $P(\mathcal{W}|W) = 1 - Q\left(\frac{\Gamma-\mu_w}{\sigma_w}\right)$ where $Q(x)$ is the error function[2] [14]. In Figure 7, we plot the curves of $P_D = P(\mathcal{C}|C)$ and $P_F = (1 - P(\mathcal{W}|W))$ as function of the threshold, Γ. Ideally, we would like to identify the threshold value which maximizes the difference between P_D and P_F. These results correspond to the experimental setup of Figure 8. We have a number of TCP traffic source-destination pairs. The link from router 1 to each TCP traffic sink has been assigned 2Mbps bandwidth and 0.01ms propagation delay. These links represent access wireless links with transmission errors. All other links are error free with 10Mbps bandwidth and 1ms propagation delay except the shared (bottleneck) wired link $0 \rightarrow 1$ whose bandwidth is 10Mbps and delay is 25ms. The buffer size at $0 \rightarrow 1$ is equal to the bandwidth-delay product and all other buffer sizes are set to default value of 50 packets. All the TCP sources and On/Off cross traffic UDP sources are started randomly between 0 sec and 3 sec and the simulations are run till 200 sec. For each cross connection, the On and Off periods are Pareto distributed with average duration of 100ms each and shape parameter of 1.5.

In this case, we have $P(C) = 2.95\%$, $P(W) = 2.5\%$, $\mu_c = 0.0744$, $\sigma_c = 0.0044$, $\mu_w = 0.0712$, $\sigma_w = 0.0075$.[3] The optimal value of Γ is found to be 0.0667 which corresponds to a misclassification penalty ratio of $\frac{R_{cw}}{R_{wc}} = 0.0787$.

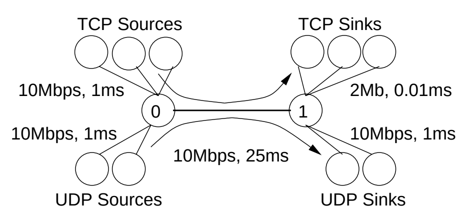


Figure 8. Wireless last-hop Topology I

6 Validation

In this section, we describe some of the tests we conducted to evaluate the characteristics of delays experienced by TCP flows in the presence of congestion and wireless losses. We conducted our experiments using the ns-2 network simulator [3]. The network topology used in the simulation is shown in Figure 9. All TCP connections traverse the links ($0 \leftrightarrow 1 \leftrightarrow 2 \leftrightarrow 3$) shared with cross traffic On/Off UDP sources as in Topology I. We assume here that the misclassification penalties, R_{cw} and R_{wc}, are equal. The rate of cross-traffic connections is controlled to induce certain $P(C)$ and $P(W)$ values.

Table 1 shows the accuracy of using empirically-obtained conditional delay distributions in Bayesian Maximum Likelihood tests. Unless otherwise specified, we used a packet trace of 800 seconds. We use this same packet trace to do both the training of the HMM to estimate the conditional delay distributions, $P(y|C)$ and $P(y|W)$, as well as to evaluate the accuracy of the classification. We notice that the Bayesian classification method even using empirical delay distributions is not perfectly accurate. The misclassification error rate is in the range $0.2 - 0.8\%$, depending on the prior values, $P(C)$ and $P(W)$,

[2] $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$.

[3] Note that if a TCP source, augmented by such Bayesian error classification, is modified to take different transmission control actions in response to different types of losses or network state, these values are likely to change.

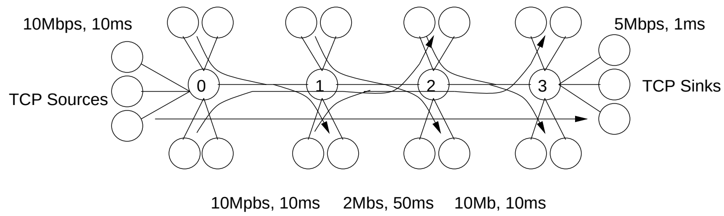


Figure 9. Wireless Last-hop Topology II

and their ratios. Furthermore, the more $P(y|C)$ and $P(y|W)$ overlap, the Bayesian classification method becomes less accurate.

Table 2 compares the accuracy of our Bayesian error classification method and that of Viterbi [13] used in [9], using conditional delay distributions estimated from a two-state HMM trained using samples from loss pairs. The HMM does not explicitly consider discretized delay values as part of its state definition (as in [9]). We observe that the Bayesian classification method performs as well while being computationally much more efficient— the computational complexity of the Bayesian method is $O(1)$ whereas that of Viterbi is $O(N^2T)$ where N is the number of states of the HMM (here N=2) and T is the number of delay samples (here the samples are coming from loss pairs).

Table 3 shows the accuracy of the Bayesian classification method using conditional delay distributions estimated from an HMM that explicitly considers discretized delay values as part of its state definition (cf. Section 3.2), trained using either all delay samples or only loss pairs. Considering all delay samples in training the HMM would be useful for delay-based estimation of the state of the communication path (as in [15]), but is bound to produce classification error when the goal is to estimate the reason of a packet loss as states that do not correspond to either congestion loss nor wireless loss get classified as such. We observe that the inaccuracy in packet-loss classification is less than 2%. Using an HMM with explicit delay component generally performs better, however at the expense of increased computational cost due to increased state space—the complexity is $O(N^2M^2T)$ as opposed to $O(N^2T)$ for a regular HMM. We also observe that in training the HMM, as expected, using all delay samples does not necessarily improve performance over using loss-pair samples only since the latter samples are more relevant to the problem of classifying congestion versus wireless losses as they capture delay properties around loss instants.

Priors		Correct Classification Prob.		P[Error]
$P(C)$	$P(W)$	$P(\mathcal{C}\|C)$	$P(\mathcal{W}\|W)$	$P(C)P(\mathcal{W}\|C)+$ $P(W)P(\mathcal{C}\|W)$
0.0196	0.0094	0.935	0.4865	0.00614
0.0081	0.0193	0.198	0.941	0.00763
0.0028	0.0287	0.510	0.971	0.0022
0.0291	0.0181	0.930	0.683	0.00777

Table 1. Accuracy of Bayesian Maximum Likelihood using Empirically-obtained Conditional Delay Distributions.

Priors		Accuracy (Bayesian)		P[Error](Bayesian)	Accuracy (Viterbi)		P[Error](Viterbi)
$P(C)$	$P(W)$	$P(\mathcal{C}\|C)$	$P(\mathcal{W}\|W)$	$P(C)P(\mathcal{W}\|C)+$ $P(W)P(\mathcal{C}\|W)$	$P(\mathcal{C}\|C)$	$P(\mathcal{W}\|W)$	$P(C)P(\mathcal{W}\|C)+$ $P(W)P(\mathcal{C}\|W)$
0.0196	0.0094	0.403	0.589	0.01556	0.41	0.649	0.01487
0.0081	0.0193	0.095	0.734	0.01246	0.75	0.458	0.01252
0.0028	0.0287	0.351	0.9	0.00475	0.77	0.525	0.01433
0.0291	0.0181	0.857	0.281	0.01723	0.696	0.449	0.01882

Table 2. Accuracy of Bayesian Maximum Likelihood and Viterbi using Conditional Delay Distributions derived from 2-state HMM trained using samples from loss pairs.

In Table 4, we evaluate the accuracy of the Viterbi algorithm using conditional delay distributions estimated from both kinds of HMM—one that is regular and the other one with hidden delay dimension in its state definition (denoted by HMM++). Both HMMs are 2-states—the former one is trained using loss pairs only and the latter one using all delay samples. We observe that the inaccuracy in packet-loss

Priors		Accuracy(all-samples)		P[Error] (all-samples)	Accuracy(loss-pairs)		P[Error] (loss-pairs)
$P(C)$	$P(W)$	$P(\mathcal{C}\vert C)$	$P(\mathcal{W}\vert W)$	$P(C)P(\mathcal{W}\vert C)+$ $P(W)P(\mathcal{C}\vert W)$	$P(\mathcal{C}\vert C)$	$P(\mathcal{W}\vert W)$	$P(C)P(\mathcal{W}\vert C)+$ $P(W)P(\mathcal{C}\vert W)$
0.0196	0.0094	0.8914	0.0524	0.01111	0.973	0.015	0.00979
0.0081	0.0193	0.089	0.996	0.0206	0	1	0.0081
0.0028	0.0287	0.000	1.000	0.0028	0	1	0.0028
0.0291	0.0181	0.987	0.004	0.0185	0.909	0.086	0.01919

Table 3. Accuracy of Bayesian Maximum Likelihood using Conditional Delay Distributions estimated from 2-state HMM with explicit delay component, trained using all samples and using loss pairs.

Priors		HMM/Viterbi			HMM++/Viterbi		
$P(C)$	$P(W)$	$P(\mathcal{C}\vert C)$	$P(\mathcal{W}\vert W)$	P[Error]	$P(\mathcal{C}\vert C)$	$P(\mathcal{W}\vert W)$	P[Error]
0.04	0.02	0.42	0.58	0.0316	0.43	0.57	0.0314
0.02	0.03	0.64	0.36	0.0264	0.59	0.55	0.0217
0.04	0.01	0.60	0.44	0.0216	0.66	0.30	0.0206
0.03	0.02	0.61	0.48	0.0221	0.66	0.36	0.0180
0.02	0.01	0.58	0.61	0.0123	0.62	0.47	0.0129

Table 4. Accuracy of Viterbi Algorithm using Conditional Delay Distributions estimated from 2-state HMM trained using loss pairs and 2-state HMM trained using all delay samples but HMM considers delay as part of the state

classification is less than 3.5%. We also observe that the performance of HMM++ with Viterbi is at least as good as HMM with Viterbi.

Priors		$N = 2$			$N = 3$			$N = 4$		
$P(C)$	$P(W)$	$P(\mathcal{C}\vert C)$	$P(\mathcal{W}\vert W)$	P[Error]	$P(\mathcal{C}\vert C)$	$P(\mathcal{W}\vert W)$	P[Error]	$P(\mathcal{C}\vert C)$	$P(\mathcal{W}\vert W)$	P[Error]
0.04	0.02	0.426	0.569	0.03158	0.384	0.671	0.03122	0.529	0.431	0.03022
0.02	0.03	0.593	0.546	0.02176	0.419	0.596	0.02374	0.439	0.560	0.02442
0.04	0.01	0.661	0.299	0.02057	0.421	0.637	0.02679	0.482	0.471	0.02601
0.03	0.02	0.656	0.375	0.02282	0.466	0.519	0.02564	0.436	0.518	0.02656
0.02	0.01	0.622	0.470	0.01286	0.322	0.743	0.01613	0.472	0.448	0.01608

Table 5. Accuracy of Viterbi using Conditional Delay Distributions derived from 2-state HMM with explicit delay component trained using all delay samples for different values of number of states

In Table 5, we observe the accuracy of Viterbi using conditional delay distributions estimated from HMM that explicitly considers delay values as part of its state definition. The HMM is trained using all delay samples for $M = 10$ and different values of N. Since using all delay samples introduces noise, increasing the number of states may not give improved packet-loss classification. But we observe that $N = 2$ works well. It is to be noted that similar experiments cannot be done to evaluate the accuracy of Bayesian Maximum Likelihood for $N > 2$ since the test is formulated as a binary (congestion vs. wireless error) decision (cf. Equation (6)).

7 Conclusion and Future Work

With the fast growth of the Internet in scope and scale, the congestion-oriented design of TCP has been challenged. Many studies have reported on the degradation in TCP performance in heterogeneous settings, and many have proposed modifications to TCP or the network itself. In this paper, we step back and examine how well can TCP estimate the error conditions of the path from its observed packet delay samples. We formulated the estimation problem as a statistical hypothesis testing and used Maximum Likelihood Ratio Tests. To infer delay distributions conditioned on loss type (congestion versus wireless), we used Hidden Markov Model together with a simple state classification heuristic (higher mean delay representing congestion state). We also examine the inclusion of discretized delay values as part of the definition of HMM state.

Our analytical and simulation results show that an efficient online error classifier can be built as long as congestion-induced losses and losses due to wireless transmission errors produce sufficiently different statistics. A simple Bayesian classification method performs as well as a Viterbi-based method. Furthermore, the explicit inclusion of delay makes the HMM-based inference more accurate at the expense of increased computational cost. Loss pairs are also found to give less noisy results than those obtained from all delay samples. Although loss pairs maybe few for short connections, a loss-type classifier maybe applied jointly to connections sharing a common network path.

In general, the TCP detector should attempt to maximize $P(\mathcal{W}|W)$ subject to a high $P(\mathcal{C}|C) \geq \alpha$. This way congestion control actions are taken in response to congestion, while avoiding a degradation in TCP throughput during wireless losses. Future work remains to investigate such loss-type-aware TCP schemes. As we also pointed out, a delay-based estimation which considers all delay samples could be used. We plan to investigate further such delay-based schemes and develop corresponding transmission control rules.

References

1. Ki Baek kim and Francois Baccelli. TCP Throughput Analysis under Transmission Error and Congestion Losses. In*Proceedings of IEEE INFOCOM*, Hongkong, March 2004.
2. *Bandwidth Used by Different Traffic Types*. `http://www.icir.org/floyd/ccmeasure.html#bandwidth`.
3. *The Network Simulator - ns-2*. `http://www.isi.edu/nsnam/ns`.
4. H. Balakrishnan, S. Seshan, and R. Katz. Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks. In *IEEE/ACM Wireless Networks, Vol 1. No. 4*, pages 10–25, December 1995.
5. Özgür B. Akan and I. F. Akyildiz. ARC: The Analytical Rate Control Scheme for Real-Time Traffic in Wireless Networks. *IEEE/ACM Transactions on Networking*, 2003.
6. L. A. Grieco and S. Mascolo. TCP Westwood and Easy RED to Improve Fairness in High-Speed Networks. In *Proceedings of Seventh International Workshop on Protocols For High-Speed Networks (PfHSN'2002)*, April 2002.
7. V. Jacobson. Congestion Avoidance and Control. In *Proceedings of ACM SIGCOMM*, August 1998.
8. D. Katabi, M. Handley, and C. Rohrs. Internet Congestion Control for High Bandwidth-Delay Product Networks. In *Proceedings of ACM SIGCOMM 2002*, August 2002.
9. J. Liu, I. Matta, and M. Crovella. End-to-End Inference of Loss Nature in a Hybrid Wired/Wireless Environment. In *Proceedings of WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2003.
10. B. A. Mah. *pchar: A Tool for Measuring Internet Path Characteristics*. `http://www.employees.org/~bmah/Software/pchar/`.
11. G. Montenegro, S. Dawkins, M. Kojo, V. Magret, and N. Vaidya. Long Thin Networks. In *RFC 2757*, Jan 1997.
12. M. Muuss. *ping*. `ftp://ftp.arl.mil/pub/ping.shar`.
13. L. Rabiner. A Tutorial on Hidden Markov models and Selected Applications in Speech Recognition. *Proceedings of IEEE, Vol. 77 No. 2*, pages 257–286, October 1989.
14. Harry L. Van Trees. Detection, Estimation and Modulation Theory, Part I¿ John Wiley and Sons, Inc., 1968.
15. Wei Wei, Bing Wang, Don Towsley and Jim Kurose. Model-based Identification of Dominant Congested Links. In *Proceedings of ACM Internet Measurement Conference*, 2003.