Modeling the short-term dynamics of packet losses

Fernando Silveira Edmundo de Souza e Silva
Federal University of Rio de Janeiro - COPPE/PESC and IM/DCC
Rio de Janeiro, Brazil
{fernando,edmundo}@land.ufrj.br

1. INTRODUCTION

Packet loss models play an essential role in computer networks analysis. Performance evaluation studies often abstract the loss and delay characteristics of a path or network with a single end-to-end analytical model. This model should be able to represent the characteristics of the path and accurately reproduce the impact of delay and losses on the studied protocol while keeping complexity low.

Likewise, there has been a growing interest in adaptive network protocols for tasks such as multimedia traffic rate control, path-switching and packet loss recovery mechanisms, to name a few. Such mechanisms must be capable of inferring future packet losses and self-adjust their behavior, in order to cope with the variability in the network conditions, aiming at achieving some given performance goals, e.g., [1], [3], [4], [10], [11]. These control mechanisms often rely on packet loss models that need to be accurate and yet simple enough for real-time analysis.

It is not uncommon to find measurements that exhibit some sort of non-stationary phenomena like long-term periodicity or trends in the average loss rate, which are difficult to model and even worse to track in real-time, [13], [3]. However, what is perhaps most noteworthy is the fact that even on channels whose statistics remain stationary over time, one can find indications of significant correlation between packets that are seconds apart, [12]. The more recent emergence of wireless technologies add to this scenario the inherent unreliability of its transmission medium, where bit error rates are ordinarily many times higher than those seen on wires, [6].

Probably the most largely applied model for packet loss processes is the 2-state Markov chain, usually referred to as the Gilbert model. Recently, the more general hidden Markov models (HMMs) have also emerged in the context of loss modeling, [7]. In the HMM proposed in [7], each hidden state is characterized by a parameter representing the individual packet loss probability. Since, in this model, hidden state transitions occur after every packet observation, the expected fraction of losses in a near future interval may converge too rapidly to the steady-state loss probability, ignoring short-term fluctuations in this measure. Nevertheless, the performance of real-time streaming applications is severely affected by these short-term fluctuations.

In this paper, we propose a variation of the basic HMM approach by constraining the model structure. Our resulting model can be seen as a hierarchical HMM, with a separate 2-state Markov chain operating inside each of the hidden chain's states. The model structure has two nice properties. First, by restricting the model we aim at reducing the overall complexity of the parameter estimation phase. Second, by assuming a specific pattern in the set of parameters, we attempt to capture the short-term dependencies in packet loss events with a 2-state Markov chain, while the longer-term dynamics is governed by a hidden Markov chain.

2. PROPOSED MODEL

A comprehensive reference on HMMs can be found in [5]. Below, we briefly present the material and notation needed for this paper. A hidden Markov model is composed of two coupled stochastic processes. The first is a Markov chain and the other is an observation process whose distribution at any given time is fully determined by the current state of the chain. In order to model packet loss events, we consider 1 as the observation symbol that indicates a loss, and 0 as a successful transmission. The work in [7] proposed a HMM with binary observations to model packet losses in a communication channel. In that model, hidden state transitions occur after every observation. We refer to this model throughout the rest of the paper as an *ordinary HMM*.

2.1 Model Definition

Suppose that transitions between hidden states occur only every S observations. Another way to think of this process is to assume that, once it enters a hidden state, it emits a *batch* of S packet transmission outcomes. For this reason, we refer to this as a *batch-observations model*. Clearly, the case of $S = 1$ is equivalent to an ordinary HMM.

This process can be modeled by a HMM in which a state can emit one of the 2^S possible observation symbols, i.e., one for each sample path for S transmission outcomes. However, this model would be unmanageable even for moderate values of S. In our approach, we restrict the distribution of observations within a batch by assuming that it is generated by a simplified Gilbert model. The reasoning behind our model is that short-term correlations could be captured by a simple process, while the dynamics at larger time scales would be governed by the hidden Markov chain. Computational savings are achieved by considering a batch of S measurements as a single observation, and computing the joint probability of the entire batch from the distribution of the generating process in each state.

Let $\{Y_t\}$ denote the underlying N-state Markov chain. The initial state distribution is given by the N-dimensional vector π, with $\pi_i = P(Y_1 = i)$. The state transition probabilities are controlled by the $N \times N$ matrix $\mathbf{A} = \{a_{ij}\}$, where $a_{ij} = P(Y_t = j|Y_{t-1} = i)$. Clearly, the constraints $\sum_{i=1}^{N} \pi_i = 1$ and $\sum_{j=1}^{N} a_{ij} = 1$ must hold. The observation process is denoted by $\{X_t\}$, where each variable, X_t, represents a vector, $[X_{t,1}, \ldots, X_{t,S}]$, with the individual outcomes for each of the S packet transmissions in the t-th batch. Also, for each hidden state we have the parameters for the 2-state Markov chain. Namely, we let:

$$r_i = P(X_{t,1} = 1|Y_t = i); \tag{1a}$$

$$p_i = P(X_{t,s} = 1|X_{t,s-1} = 0,\, Y_t = i), \quad 1 < s \leq S; \tag{1b}$$

$$q_i = P(X_{t,s} = 0|X_{t,s-1} = 1,\, Y_t = i), \quad 1 < s \leq S. \tag{1c}$$

We refer to the set of parameters as the tuple $\lambda = (\pi, \mathbf{A}, \mathbf{r}, \mathbf{p}, \mathbf{q})$, where $\mathbf{r}$, $\mathbf{p}$, and $\mathbf{q}$ are vectors containing the respective r_i, p_i and q_i, for each state i.

2.2 Parameter Estimation

We derive a set of formulae for iterative estimation of model parameters from a trace of packet loss measurements, based on the Expectation-Maximization (EM) technique, [2], [5]. In what follows, we consider that the packet measurements are segmented in T sets of size S. More specifically, let x_t denote the vector of measures, $[x_{t,1}, \ldots, x_{t,S}]$, with the outcomes of each packet in the t-th batch.

Whenever there is no ambiguity, we will be using the abbreviated form, $X_{i:j}$ (and accordingly $Y_{i:j}$), to denote the compound event that $X_i, X_{i+1}, \ldots, X_j$, $(Y_i, Y_{i+1}, \ldots, Y_j)$ assume the respective values $x_i, x_{i+1}, \ldots, x_j$ $(y_i, y_{i+1}, \ldots, y_j)$. In the particular case where $i = j$, we will simply be writing X_i (or equivalently Y_i). On the other hand, we will use $\mathbf{X}$ (and $\mathbf{Y}$) when the subindices span the full range from 1 to T, i.e., $X_{1:T}$ $(Y_{1:T})$.

It is important to notice that, in order to evaluate the likelihood of a sample, one does not need to record the individual packet measurements. It is enough to keep track of the *sufficient statistics*[1] denoted, in each batch of measures, x_t, as:

$$x_{t,1} = \text{outcome of the first packet in } x_t, \tag{2a}$$
$$S_t^{ij} = \text{\# of transitions i} \rightarrow \text{j in } x_t,\ i,j \in \{0,1\}. \tag{2b}$$

Given an instance of x_t, we are interested in computing the probability that $X_t = x_t$, given the hidden state, $Y_t = y_t$. Using the statistics defined above, we denote as b_{y_t,x_t}, the probability of observing a full batch of observations x_t using the parameters in the hidden state given by y_t. Clearly, we have:

$$b_{y_t,x_t} = r_{y_t}(p_{y_t})^{S_t^{01}}(1-p_{y_t})^{S_t^{00}}(q_{y_t})^{S_t^{10}}(1-q_{y_t})^{S_t^{11}}, \tag{3}$$

whenever $x_{t,1} = 1$, or equivalently:

$$b_{y_t,x_t} = (1-r_{y_t})(p_{y_t})^{S_t^{01}}(1-p_{y_t})^{S_t^{00}}(q_{y_t})^{S_t^{10}}(1-q_{y_t})^{S_t^{11}}, \tag{4}$$

if $x_{t,1} = 0$.

We also define the following probability measures, using the notation from [5]:

$$\alpha_t(i) = P(X_{1:t}, Y_t = i|\lambda), \tag{5a}$$
$$\beta_t(i) = P(X_{t+1:T}|Y_t = i, \lambda), \tag{5b}$$
$$\gamma_t(i) = P(Y_t = i|\mathbf{X}, \lambda), \tag{5c}$$
$$\xi_t(i,j) = P(Y_t = i, Y_{t+1} = j|\mathbf{X}, \lambda), \tag{5d}$$

where $\alpha_t(i)$ and $\beta_t(i)$ are calculated through the forward-backward recursions (see [5] for details), and the following identities can be established:

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}, \tag{6a}$$

$$\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_{jx_{t+1}}\beta_{t+1}(j)}{\sum_{k=1}^{N} \alpha_t(k)\beta_t(k)}. \tag{6b}$$

The joint likelihood of hidden states and observations given some parameter estimates, λ, is defined as:

$$P(\mathbf{X}, \mathbf{Y}|\lambda) = \pi_{y_1} b_{y_1,x_1} \prod_{t=2}^{T} a_{y_{t-1},y_t} b_{y_t,x_t}. \tag{7}$$

[1]A statistic $\Gamma(X)$ is a *sufficient statistic* for θ if the distribution of the sample X given the value of $\Gamma(X)$ does not depends on θ.

The EM approach consists of alternating between two steps. The Expectation step (or E-step) evaluates the auxiliary function, $Q(\lambda|\overline{\lambda})$, with respect to λ, making use of the current parameter estimates, $\overline{\lambda}$. This function has the form:

$$Q(\lambda|\overline{\lambda}) = \sum_{\forall \mathbf{y}} \log P(\mathbf{X}, \mathbf{Y}|\lambda) P(\mathbf{Y}|\mathbf{X}, \overline{\lambda}). \tag{8}$$

The Maximization step (or M-step) of EM obtains λ in order to maximize $Q(\lambda|\overline{\lambda})$. It is easily shown that this procedure generates a sequence of estimates with non-decreasing likelihood, [2]. Using (7), Equation (8) can be broken in independent terms as:

$$\begin{aligned} Q(\lambda|\overline{\lambda}) = &\sum_{i=1}^{N} \log \pi_i \gamma_1(i) + \\ &\sum_{i=1}^{N}\sum_{j=1}^{N} \log a_{ij} \sum_{t=1}^{T-1} \xi_t(i,j) + \\ &\sum_{i=1}^{N} \log r_i \sum_{t=1}^{T} \mathbb{I}\{x_{t,1} = 1\}\gamma_t(i) + \\ &\sum_{i=1}^{N} \log(1-r_i) \sum_{t=1}^{T} \mathbb{I}\{x_{t,1} = 0\}\gamma_t(i) + \\ &\sum_{i=1}^{N} \log p_i \sum_{t=1}^{T} S_t^{01}\gamma_t(i) + \\ &\sum_{i=1}^{N} \log(1-p_i) \sum_{t=1}^{T} S_t^{00}\gamma_t(i) + \\ &\sum_{i=1}^{N} \log q_i \sum_{t=1}^{T} S_t^{10}\gamma_t(i) + \\ &\sum_{i=1}^{N} \log(1-q_i) \sum_{t=1}^{T} S_t^{11}\gamma_t(i). \end{aligned} \tag{9}$$

Maximizing each term of (9) and bearing in consideration the stochastic constraints on the parameters, leads to estimation formulae for π and $\mathbf{A}$, which are identical to those of a traditional HMM, [5]:

$$\pi_i = \gamma_1(i), \tag{10a}$$

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}, \tag{10b}$$

and the specific formulae for the observation parameters of the batch-observations model:

$$r_i = \frac{\sum_{t=1}^{T} \mathbb{I}\{x_{t,1} = 1\}\gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)}, \tag{11a}$$

$$p_i = \frac{\sum_{t=1}^{T} S_t^{01}\gamma_t(i)}{\sum_{t=1}^{T}(S_t^{01} + S_t^{00})\gamma_t(i)}, \tag{11b}$$

$$q_i = \frac{\sum_{t=1}^{T} S_t^{10}\gamma_t(i)}{\sum_{t=1}^{T}(S_t^{10} + S_t^{11})\gamma_t(i)}. \tag{11c}$$

2.3 Computational Costs

For a training sample of size T, an ordinary HMM with N states performs a number of operations proportional to N^2T, in order to evaluate the forward and backward variables, [5]. These calculations are the dominant factor of complexity in the parameter estimation procedure.

The computational advantage of our model is evident from equations (11), since each training iteration depends only on the statistics defined in (2). Since the whole training sample can be compactly described by these sufficient statistics, each iteration is faster by a factor of S. Also, since the measurements used in training are usually done at the receiving side and need to be sent back to the transmitter, there is also a smaller overhead in application payload due to the out-of-band data required for model estimation.

3. NUMERICAL EXAMPLE

We performed an experiment to assess the ability of our model to predict future packet loss performance in an on-line task, compared to the same kind of predictions provided by an ordinary HMM. This experiment used a set of traces generated between academic sites in Brazil and in the United States. The traffic pattern of these traces emulates a simplified VoIP tool, sending 50 packets per second with 324 bytes per packet.

Every 5 seconds, for both models, we evaluate the expected loss fraction in the next 5 seconds, given the outcomes of the packet losses in the last 10 seconds. An algorithm that evaluates this measure is the subject of an extended version of this paper, [8]. In addition, model parameters are re-estimated every 3 minutes, using the sample of measurements from the latest 3 minutes. For the batch-observations HMM, we used $S = 50$ packets, so that hidden state transitions occur every second. Moreover, both models considered have 10 hidden states.

Figure 1 shows the predictions of both models compared to the original measurements, for a sample trace of 30 minutes. It can be noticed that the ordinary HMM was not capable of reproducing the variations in the loss rate as well as our proposed model. We attribute this difference to the fact that, in the former model, the hidden state, along with channel statistics, may change after each packet transmitted. As a consequence, steady state is reached much faster than for the latter model, in which hidden state transitions occur at a longer time scale, i.e., 50 packets in this case.

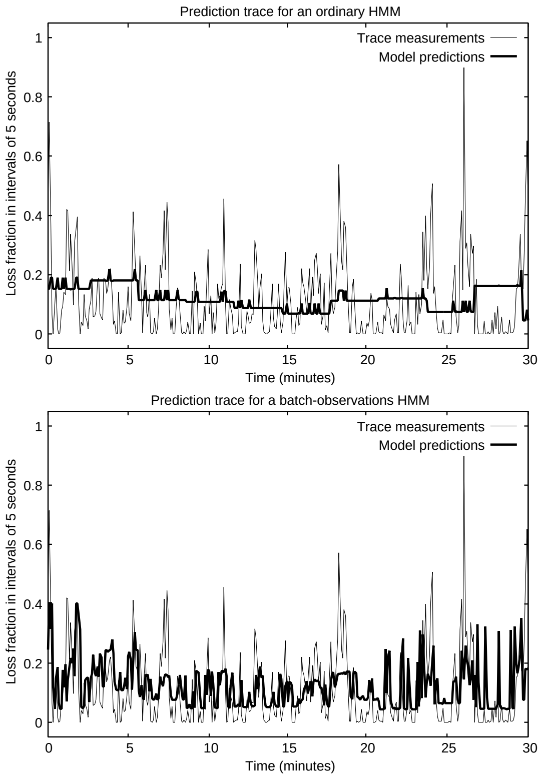

Figure 1: Prediction results for an ordinary HMM and for the batch-observations model.

4. CONCLUSIONS AND FUTURE WORK

We developed a hierarchical loss model, aimed at capturing short-term variations in channel statistics. The parameters of this model can be estimated from a trace of measurements, using equations that are presented in this paper. Through experiments that used real Internet loss traces, we have found that our model has advantages over a traditional hidden Markov model, in terms of both prediction accuracy and computational complexity. In subsequent work, [9], we have applied our model's predictions to the problem of dynamically selecting forward error correction in an interactive streaming application with encouraging results.

5. REFERENCES

[1] J. C. Bolot, S. F. Parisis, and D. F. Towsley. Adaptive FEC-based error control for Internet telephony. In *Proceedings of the IEEE INFOCOM*, pages 1453–1460, March 1999.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from imcomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[3] F. P. Duarte, E. A. de Souza e Silva, and D. F. Towsley. An adaptive FEC algorithm using hidden Markov chains. *SIGMETRICS Performance Evaluation Review*, 31(2):11–13, September 2003.

[4] M. Karol, P. Krishnan, and J. J. Li. Rapid fault detection and recovery for IP telephony. In *Proceedings of IEEE ICC*, volume 3, pages 1478–1483, March 2004.

[5] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, February 1989.

[6] T. S. Rappaport. *Wireless Communications: Principle and Practice*. Prentice Hall, 2001.

[7] K. Salamatian and S. Vaton. Hidden Markov modeling for network communication channels. In *Proceedings of the ACM SIGMETRICS*, pages 92–101, June 2001.

[8] F. Silveira and E. de Souza e Silva. A method for predicting packet losses with applications to continuous media streaming. In *Proceedings of the Brazilian Symposium on Computer Networks (SBRC)*, June 2006.

[9] F. Silveira, E. H. Watanabe, and E. de Souza e Silva. Adaptive forward error correction for interactive streaming over the Internet. In *Proceedings of the IEEE GLOBECOM*, November 2006.

[10] S. Tao and R. Guerin. On-line estimation of Internet path performance: an application perspective. In *Proceedings of IEEE INFOCOM*, volume 3, pages 1774–1785, March 2004.

[11] S. Tao, K. Xu, A. Estepa, T. Fei, L. Gao, R. Guerin, J. F. Kurose, D. F. Towsley, and Z. Zhang. Improving VoIP quality through path switching. In *Proceedings of IEEE INFOCOM*, volume 4, pages 2268–2278, March 2005.

[12] M. Yajnik, S. B. Moon, J. F. Kurose, and D. F. Towsley. Measurement and modeling of the temporal dependence in packet loss. In *Proceedings of the IEEE INFOCOM*, pages 345–352, March 1999.

[13] Y. Zhang, V. Paxson, and S. Shenker. The stationarity of Internet path properties: routing, loss, and throughput. Technical report, ACIRI, May 2000.