Markov-based Trace Analysis Algorithm for Wireless Networks

by

Orororo Oghenero Stephen

Abstract - A model and simulation channel need to be built on understanding better and testing the network protocol and application behavior. In this project, a two-state Markov based on Gilbert's algorithm model can build a model based on the stationarity assumption of data and a discrete-time Markov chain (DTMC) that factors in the non-stationarity of a wireless network trace in a short period. These Markov Models are used to investigate and develop the statistical characteristics of the underlying wireless network performance, like throughput, packet loss, and latency, to understand the behavior of wireless data transmissions better.

Introduction

Understanding the network protocol and application behavior requires accurate techniques which model and simulate network channels; such techniques play a vital role in understanding and behavior of the Network. Every day, network protocol and design are being changed. Also, the designing of communication protocols rises in involvement. [2] To estimate Network's performance, we should be aware of the techniques such as simulation, analysis of actual data, and analysis of other models. Due to the better understanding of network behavior and structure of communication protocol to modeling network, we need to look at the error behavior at the Link and upper layers. A deep understanding of packet failure techniques and error burstiness is essential for designing and monitoring error control protocols. [3]

In our everyday life, we use applications that stream audio and video. Those applications will work better when they benefit from a better underrating of network behavior. For example, in the call application, we can benefit from the call quality without having latency, glitter, and lags via estimations of conditions in real-time ways and networks. [5] The current network model approach to the error modeling- Gilbert model does not entirely fit the current network wireless network due to the requirement of stationary traces over a short period. And the nature of the wireless network due to multipath fading or shadowing causes the trace to be non-stationary. This can cause the traditional stochastic analysis of wireless traces is likely to be less accurate than the required accuracy for use in real life. And the use of an inaccurate analytical model resulted in inferior error control protocol parameter choices. Thus, a new algorithm must be presented to improve the model.

Related works

There is significant interest in using network measurements to model network behavior. However, very few researchers address the problem of non-stationarity in network modeling. Zhang and others study stationarity on the Internet and introduce a new notion of stationarity that is more relevant to network properties [16]. They call a dataset operationally stationary if the statistics of interest remain within bounds considered operationally equivalent. Their most exciting finding is that stationarity depends on the time scale used for evaluation. Others have looked at the stationarity behavior of network traffic, traffic stationarity. For example, Molnar and Gefferth [11] propose a simple approach for identifying stationary intervals and analyzing them independently. They introduce a new technique for identifying these intervals. Leland et al. [8] study the stationarity of self-similar models of network traffic.

Researchers have applied traditional models to analyze non-stationary data collected in computer networks. In particular, they have used traditional models to characterize the loss process of various channels. Bolot et al. [3] use a characterization of the loss process of audio packets to determine the appropriate error control scheme for streaming audio. They model the loss process as a two-state Markov chain and show that the loss burst distribution is approximately geometric. Yajnik et al. [15] characterize the packet loss in a multicast network by examining the spatial (across receivers) and temporal (across consecutive packets) correlation in packet loss. Their modeling of temporal loss using a 3rd-order Markov chain is of particular interest. Yajnik's work identifies the problem of non-stationarity in their datasets, and they analyze the data by removing these parts of the data that experience nonstationary error behavior. There is also related work in wireless traffic modeling. Nguyen et al. [12] present a two-state Markov wireless error model (i.e., Gilbert model) and develop an improved model based on collected Lucent 900 MHzWaveLAN error traces. Building on this work, Balakrishnan and Katz [1] also contained error traces from a Lucent 900 MHzWaveLAN network and developed a two-state Markov chain error model. Willig et al. [14] present a special class of Markov models called bipartite. Zorzi and Rao [17] also investigate the error characteristics of a wireless channel and compare an Independent and Identically Distributed (IID) model to the Gilbert model. Their work postulates that higher-order models are not necessary.

System Model and Simulation

The following data were considered for the Simulation Environment.

- modulation scheme: QPSK, BPSK, 16QAM, and 64QAM
- Signal Noise Ratio (SNR) From 15db to 30db
- Distance between transmitter and receiver over 10 meters

- bandwidth channel considered between 20MHZ to 40MHZ
- Traffic parameters: packet size 1000bytes, Data rate Kbps 100000

all simulations were carried out in MATLAB.

Network performance considered

Latency: Network latency is the delay in network communication. It shows the time that data takes to transfer across the network. Networks with a longer delay or lag have high latency, while those with fast response times have low latency. [6]

Throughput: network throughput is the amount of data moved successfully from one place to another in a given period. Network throughput is typically measured in bits per second (bps)

Packet Loss: Packet loss describes lost packets of data not reaching their destination after being transmitted across a network. Packet loss occurs when network congestion, hardware issues, software bugs, and several other factors cause dropped packets during data transmission.

simulation result

NODE ID	LATENCY(BSPK)	LATENCY(QPSK)
1	4	4.9
2	3.15	4.4
3	3	4
4	2.64	3.5
5	2.32	2.9
6	2.1	2.45

Table 1: compares results between BSPK and QPSK for Latency

DOWNLOAD STATIONS	THROUGHPUT (64QAM)	THROUGHPUT (16QAM)
1	23.3	20.4
2	36.50	32.0
6	47.0	43.2
8	60.1	57.6
9	69.2	63.3
15	111.3	105.7
20	150.6	137.7
25	188.3	168.2
30	221.15	197.1
40	288.4	250.0
50	360.3	304.4
60	423	346.1
74	498	403.2
100	509.14	414.4
140	509.19	414.56

Table 2: compares results between 64QAM and 16QAM for throughput

NODE ID	PACKET LOSS(BSPK)	PACKET LOSS(QPSK)
1	0.2	0.198
2	0.18	0.175
3	0.15	0.171
4	0.143	0.139
5	0.13	0.12
6	0.11	0.09

Table3: compares results between BSPK and QPSK for Packet loss

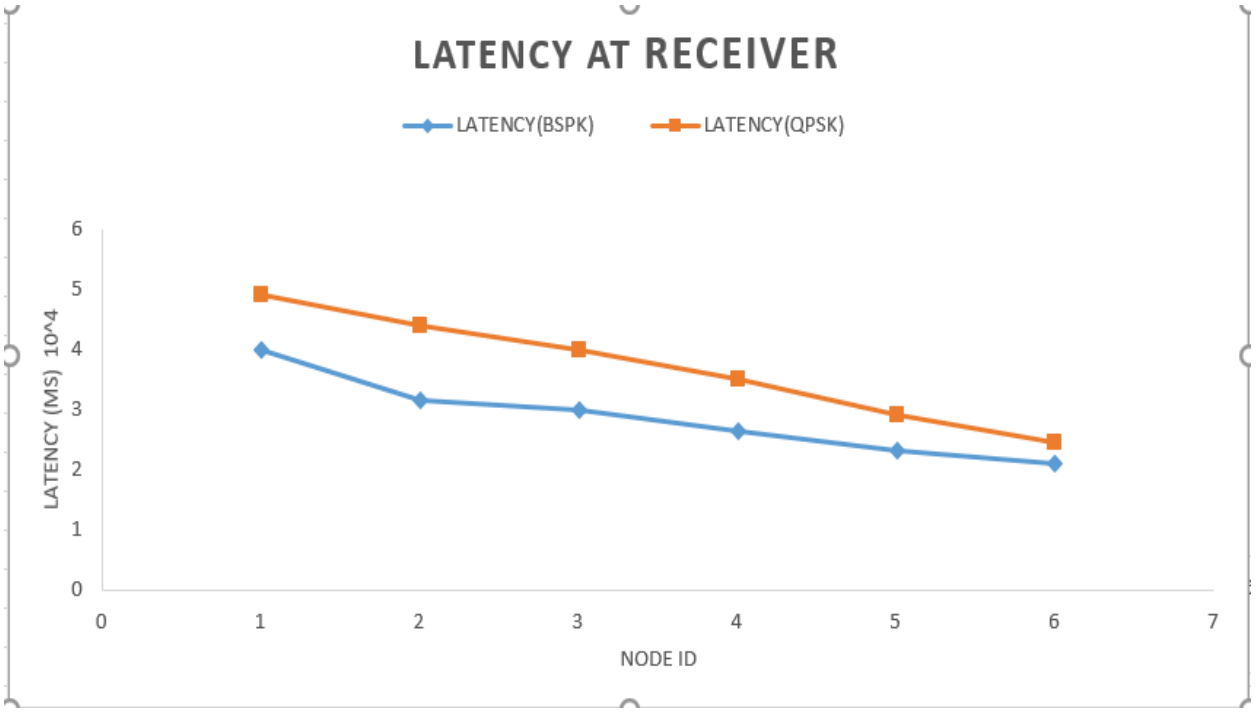

Figure1: compares results between BSPK and QPSK for Latency

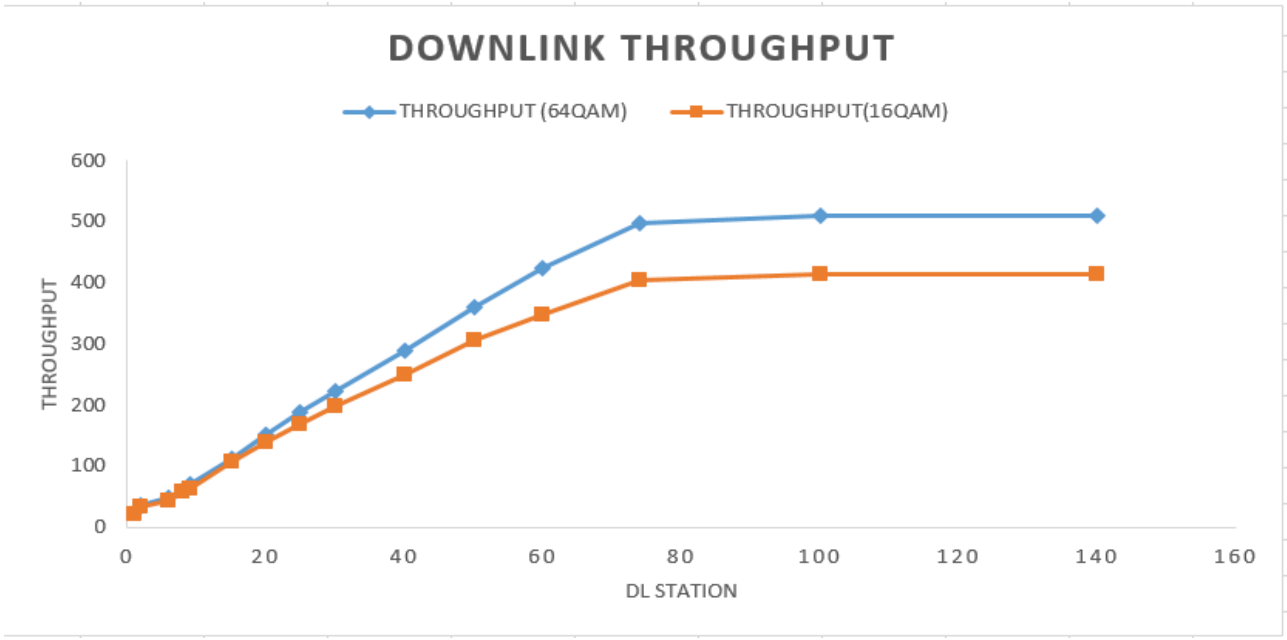

Figure2: compares results between 64QAM and 16QAM for throughput

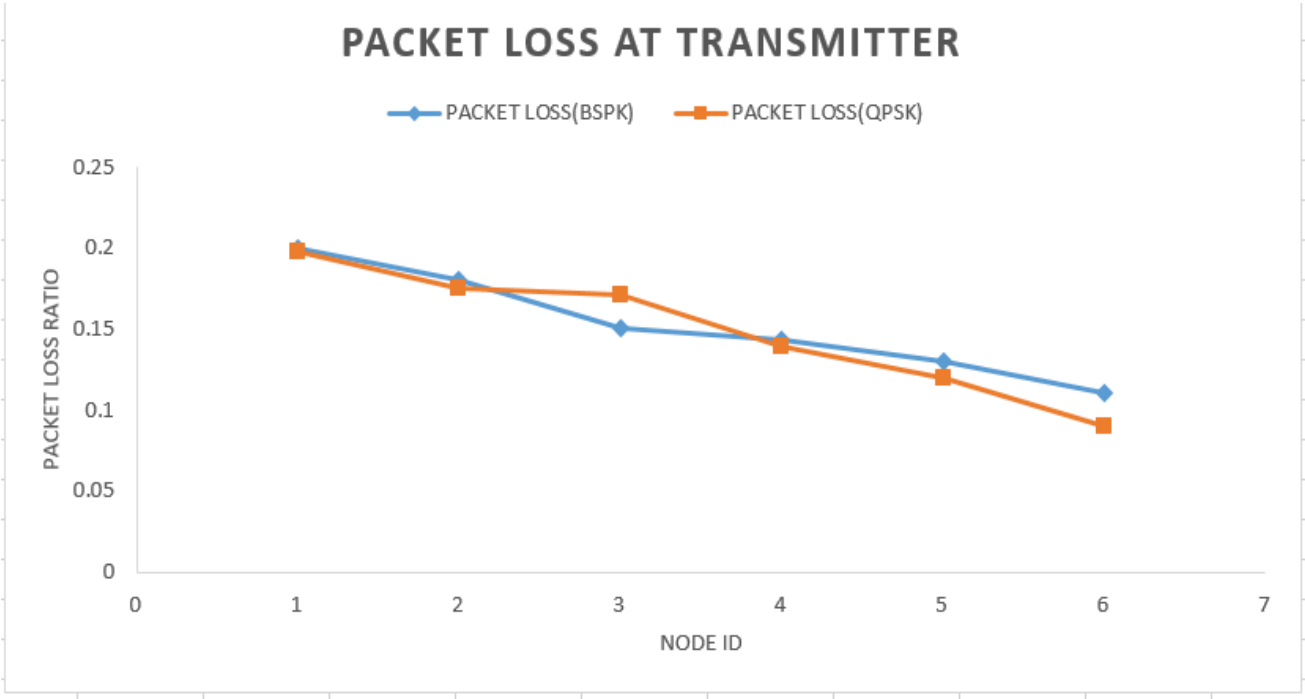

Figure3: compares results between BSPK and QPSK for Packet loss

Theoretical Background of the proposed Markov model (Gilbert)

In this model, there are two states: good states and bad states. The mean statistical value for the data set is calculated, and it is assumed to be in the good state; if the value of the data set is equal to or higher than the calculated mean value, it is considered to be in the bad state if the value is lower than the mean. The following set represents the state: S = {G B}. the set of the transition matrix is expressed by

$$S = [S_t, S_{t+1}, S_{t+2} \cdots , S_{t+k}]$$

At the initial state, the channel might be in a good state or a bad state. At the transition of a new state for a new bit, it will change to a new state or remain in the same state.

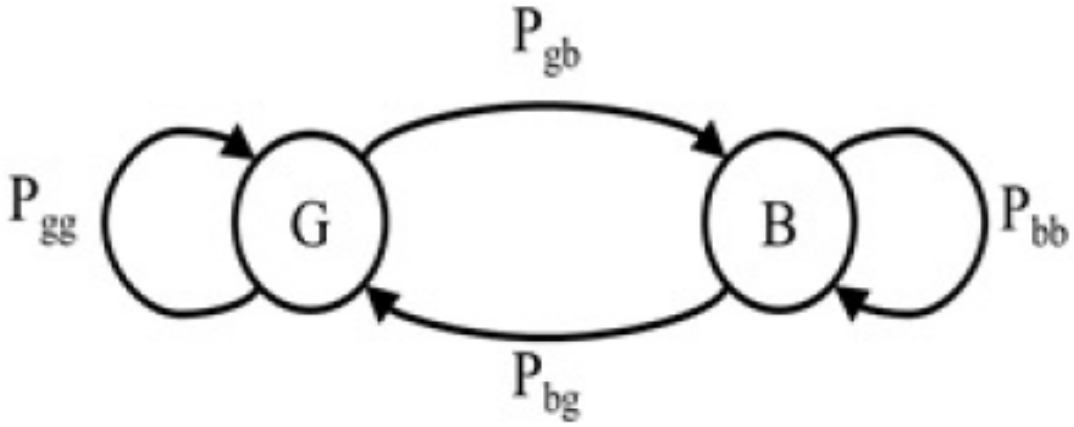

Figure 4

Transition Matrix

$$A_{(t)} = \begin{bmatrix} Pgg & Pgb \\ Pbg & Pbb \end{bmatrix}$$

The four transition probabilities are as follows:

$$P_{gg}(t) = \Pr\{S_{t+1} = g | St = g\}$$
$$P_{gb}(t) = \Pr\{S_{t+1} = b | St = g\}$$
$$P_{bb}(t) = \Pr\{S_{t+1} = b | St = b\}$$
$$P_{bg}(t) = \Pr\{S_{t+1} = g | St = b\}$$

We define Π_t as the state probability distribution at time t.

$$\Pi_t = [\pi_{t,g} \pi_{t,b}],$$

Where is the steady-state vector that expresses the total percentage of a state in a Markov chain, this vector can be computed by raising P to immense power.

Two-State Markov Model

From the transition diagram in fig4 above, P_{gg} represents moving from a good state and remaining in a good state, P_{gb} represents moving from a good state to a bad state, and P_{bb} represents moving from a bad state and staying in a bad state. P_{bg} represents moving from a bad state to a good state.

For modeling, the state was classified as good or bad by computing the mean of the result obtained to determine a threshold. For throughput, the result obtained, any data below the threshold were regarded as being in a bad state, and data above or equal to the threshold was

classified as in a good state. for latency, the threshold was determined by computing the mean. Any data below the threshold were classified as a good state, and any data above the threshold were classified to be in a bad state; for packet loss ratio, any data below the threshold was considered to be in a good state, and any data above the threshold were classified to be in a bad state.

To statistically evaluate the performance measure of the wireless network, the system's state at the equilibrium or steady state needs to be computed. Below is the performance measure at steady state.

Transition Matrix for throughput

$$\text{Transition Matrix for Throughput} \begin{bmatrix} \frac{6}{15} & \frac{6}{15} \\ \frac{9}{15} & \frac{9}{15} \end{bmatrix}$$

$$\text{the initial state distribution } s_0 = \begin{bmatrix} \frac{6}{15} & \frac{9}{15} \end{bmatrix}^t$$

$$\text{steady-state distribution} = \begin{bmatrix} \frac{2}{5} & \frac{3}{5} \end{bmatrix}^t$$

Transition Matrix for latency (at Receiver)

$$\text{Transition Matrix for Latency} \begin{bmatrix} \frac{1}{2} & \frac{1}{6} \\ \frac{1}{2} & \frac{5}{6} \end{bmatrix}$$

$$\text{the initial state distribution } s_0 = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \end{bmatrix} t$$

$$\text{steady-state distribution} = \begin{bmatrix} \frac{1}{4} & \frac{3}{4} \end{bmatrix} t$$

Transition Matrix for packet-loss (at Transmitter)

$$\text{transition Matrix for Packet-loss} \begin{bmatrix} \frac{2}{3} & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} \end{bmatrix}$$

$$\text{the initial state distribution } s_0 = \begin{bmatrix} \frac{2}{3} & \frac{1}{3} \end{bmatrix} t$$

$$\text{steady-state distribution} = \begin{bmatrix} \frac{3}{5} & \frac{2}{5} \end{bmatrix} t$$

Discrete-Time of Markov model

the second Markov model that factors in the non-stationarity of a wireless network trace. process {Xn | n 0} that takes values in a discrete space E A DTMC is defined by its memory and its transition probabilities and is characterized as follows:

$$\Pr(X_{n+1} = j \mid X_0 = i_0, X_1 = i_1, \ldots, X_n = i_n)$$
$$= \Pr(X_{n+1} = j \mid X_{n-z+1}, 1 \leqslant z \leqslant K), \quad [10]$$

K defines the memory of the DTMC. To calculate the DTMC's memory, conditional entropy is needed to find the order of the Markov Chain. Given the prior history, conditional entropy can indicate the randomness of the next piece in a trace. And the conditional entropy is calculated by the following formula. [2]

$$H(i) = -\sum_{\vec{x}} \frac{\xi(\vec{x})}{T_{\text{samples}}} \sum_{y \in \{0,1\}} \frac{\xi(y, \vec{x})}{\xi(\vec{x})} \log_2 \frac{\xi(y, \vec{x})}{\xi(\vec{x})}. \quad [9]$$

Stationarity

The DTMC algorithm provides lossy and error-free states and parameterized transitions between them as a function of a fixed parameter called the change-of-state constant C. The change-of-state constant C is the mean plus one standard deviation of a trace's length of error bursts. The value of C determines the threshold for computation. The next step was to remove trace sections consisting of error-free bursts of length equal to or greater than C. doing that, to ensure the processed result will have stationarity error statistic properties. [1]

Once the lossy sub-trace has been generated and confirmed as a stationary process, the next step was to model the lossy sub-trace as a DTMC with memory k. The memory k can impact the complexity of the model, and determining the right k can ensure the model has the right level of complexity while not having a significant impact on the accuracy (where k is the entropy order). [11] And lower entropy means the model will be more accurate. In the previous gilbert model, I used the order K as one, so the entropy is the highest, and accuracy is the lowest.

The application of the DTMC algorithm to input trace can be summarized as follows.

Calculate the mean (*m*e) and standard deviation (*sd*e) values for error burst lengths in the trace.

2. Set *C*, the change-of-state constant, equal to (*m*e + *sd*e).

3. Partition the trace into lossy state and error-free state portions using the following definitions:

• *Lossy state*: runs of 1's and 0's, with the first element being a 1, and runs of 0's that have lengths less than or equal to the *C*.• *Error-free state*: runs of 0's that have lengths greater than *C*.

4. Create a lossy subtrace from the lossy state portions of the error trace.

5. Model lossy subtrace as a DTMC and calculate its order

and transition probabilities. Determine the best-fitting distributions of the length

$$S_{\text{error}}(\vec{y}, \vec{x}) = \sqrt{\left[\frac{1}{n(n-2)}\right]\left[f(y) - \frac{\left[n\sum xy - \sum x \sum y\right]^2}{f(x)}\right]} \quad [17]$$

The standard error was calculated for Bn=0.013 and Gn=0.025. The smaller normal error value means a more accurate prediction.

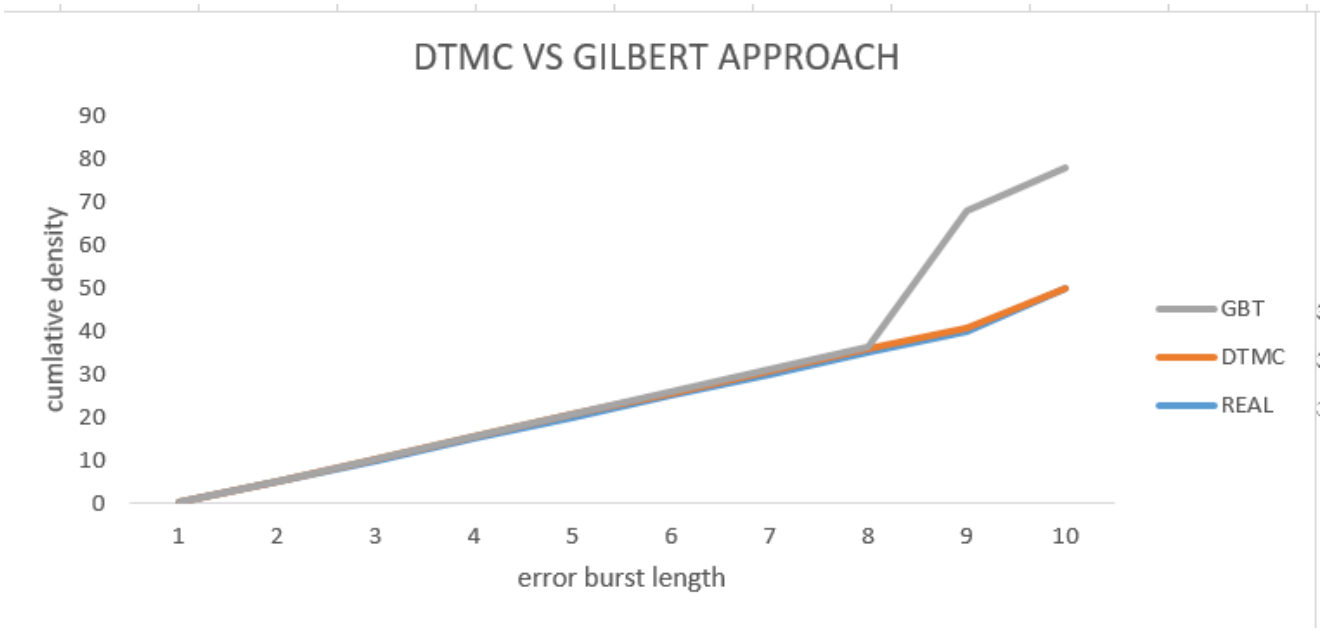

Figure 5

CONCLUSION

This paper uses two Markov-based models to build a network modeling approach to error modeling. The traditional two-state Gilbert model was helpful for the traditional network, with error statistics remaining relatively constant over a short period. At a steady state, we could statically compute various wireless performance measures like throughput, latency, and packet error ratio at the transmitter and receiver. But due to the increased use of the wireless network, the error statistics in the network environment are no longer stationary. Therefore, we must clean

the data and create a stationary process from a non-stationary dataset. Once the data have been convincing to the stationary process, we need to find the right order of the Markov chain to analyze the model. as the randomness of the wireless network increases, as seen in fig 5, the gilbert model was not as accurate in comparison to the DTMC model approach.

Reference

[1] S. Floyd, E. Kohler, Internet research needs better models, in: Hotnets-I, October 2002.

[2] A. Konrad, B. Zhao, A. Joseph, R. Ludwig, A Markov-based channel model algorithm for wireless networks, in: Proceedings of the Fourth ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), 2001.

[3] R. Ludwig, A. Konrad, A. Joseph, Optimizing the end-to-end performance of reliable flows over wireless link, in: Proceedings of ACM/IEEE MobiCom, 1999.

[4] Y. Zhang, V. Paxson, S. Shenker, The stationarity of internet path properties: Routing, loss, and throughput, in: ACIRI Technical Report, 2000.

[5] S. Molnar, A. Gefferth, On the scaling and burst structure of data traffic, in: 8th International Conference on Telecommunication Systems, Modelling and Analysis, 2000.

[6] W. Leland, M. Taqqu, W. Willinger, D. Wilson, On the self-similar nature of ethernet traffic, in: IEEE/ACM Transaction on Networking, 1994.

[7] J. Bolot, S. Fosse-Parisis, D.Towsley, Adaptive FEC-based error control for internet telephony, in: Proceedings of Infocom, ACM, 1999.

[8] M. Yajnik, J. Kurose, D. Towsley, Packet loss correlation in the MBone multicast network: Experimental measurements and Markov chain models, UMASS COMPSCI Technical Report 95-115.

[9] G. T. Nguyen, R. Katz, B. Noble, A trace-based approach for modeling wireless channel behavior, in: Proceedings of the Winter Simulation Conference, 1996, pp. 597–604.

[10] H. Balakrishnan, R. Katz, Explicit loss notification and wireless web
[10] H. Balakrishnan, R. Katz, Explicit loss notification and wireless web
performance, in: Proceedings of the IEEE Globecom Internet Mini-Conference, 1998.

[11] A. Willig, M. Kubisch, A. Wolisz, Measurements and stochastic modeling of a wireless link in an industrial environment, in: Technical Report TKN-01-00, Technical University Berlin, 2001.

[12] M. Zorzi, R. R. Rao, On the statistics of block errors in bursty channels, in: IEEE Transactions on Communications, 1997.

[13] J. Bendat, A. Piersol, Random Data: Analysis and Measurement Procedures, John Wiley and Sons, 1986.

[14] B. Kedem, Binary Time Series, New York and Basel, 1980.

[15] I. L. MacDonald, W. Zucchini, Hidden Markov and Other Models for Discretevalued Time Series, 1st Edition, Chapman & Hall/CRC, 1997.

[16] J. Pearl, Probabilistic Reasoning in Intellegent Systems: Networks of Plausible Inference, Morgan Kaufmann, 1988.

[17] R. Jain, The Art of Computer Systems Performance Analysis, John Wiley and Sons, 1991.