Modeling and Simulation of Wireless Link Quality (ETT) Through Principal Component Analysis of Trace Data

Anh Le, Prashant Krishnamurthy, David Tipper and Konstantinos Pelechrinis
Graduate Telecommunications and Networking Program, University of Pittsburgh
135 N. Bellefield Avenue, Pittsburgh, PA 15260
{atl13, prashk}@pitt.edu, tipper@tele.pitt.edu, kpele@sis.pitt.edu

ABSTRACT

Principal Component Analysis (PCA) is a powerful method in data analysis. In this paper, we employ the capabilities of PCA combined with statistical fits to trace data to develop tractable models that can be used to simulate the quality of links in wireless mesh networks using the expected transmission time (ETT) metric. We apply principal component analysis to ETT traces from a wireless mesh network to determine what features in the ETT traces are important and to extract any meaningful relationships therein. We demonstrate that PCA can be used to efficiently approximate large volumes of ETT values. In particular, the ETT trace for each link can be expressed as a combination of two basis vectors – one fairly stable and the other containing the variations in time. We also show how the extracted features can be employed to simulate ETT for a given network topology with and without known ETT trace data.

Categories and Subject Descriptors

I.6.5 [**SIMULATION AND MODELING**]: Model Development—*Modeling methodologies*

General Terms

Algorithms, Design, Measurement

Keywords

Principal Component Analysis, Expected Transmission Time

1. INTRODUCTION

The Expected Transmission Time (ETT) for transmitting a packet over a link has been used as a metric of wireless link quality in mesh networks for the last several years [1]. ETT is time varying and is derived from the Expected Transmission Count (ETX) [2] as follows: ETT = $\frac{S}{B} \times$ ETX, where S and B are packet size and link bandwidth respectively. ETX, in turn, is calculated as $ETX = 1/(1-p)$, where p is the probability of *unsuccessful* transmission over a link. The value of p itself is determined as $p = 1-(1-p_f)(1-p_r)$, where p_f and p_r are the probabilities of failure on the forward and reverse directions over the link respectively. The values of p_f and p_r depend on the channel quality, distance between nodes, interfering transmissions, collisions, traffic distribution and flows. It is possible to simulate the value of ETT by assuming some error distribution for bits or packets on a given link. However, the value of ETT is more complex due to the dynamic factors that impact it. While the ETT value is often used to test the performance of routing schemes and more recently for resilience analysis [3], we are not aware of any reasonable models to simulate the ETT values directly without a detailed link and network model simulation. Here, we employ trace data for ETT values in real networks to see whether they can be used to develop models for simulating the ETT values.

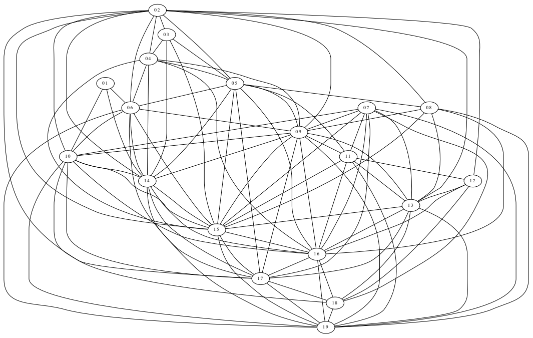

Figure 1: Logical network topology

We develop an efficient way to extract the dynamic features from ETT traces by utilizing principal component analysis and use such features to develop a general model that can be used to simulate the ETT value. The structure of the paper is as follows. In section 2 we describe the data and the original source of the data. Section 3 provides some background and related work. In section 4 we present the approach used to analyze the trace data and show numerical results. Section 5 shows how we can use the analysis to simulate ETT values. Finally, we point out some limitations of our work and suggest future work in section 6.

2. TRACE DATA AND USAGE

The ETT datasets were collected by the authors of [4] on UCSB's MeshNet and were used to test the design of a new

PE-WASUN'11, November 3–4, 2011, Miami, Florida, USA.

Table 1: Neighbors list

Node	Deg.	Neighbors
01	3	06 10 14
02	12	03 04 05 06 08 09 10 12 13 14 15 17
03	5	02 04 05 14 15
04	8	02 03 05 06 09 14 15 16
05	11	02 03 04 06 08 09 11 14 15 16 17
06	10	01 02 04 05 09 10 14 15 17 19
07	9	09 10 11 13 15 16 17 18 19
08	7	02 05 09 13 15 16 19
09	14	02 04 05 06 07 08 10 11 13 14 15 16 17 19
10	11	01 02 06 07 09 14 15 16 17 18 19
11	8	05 07 09 12 13 15 16 19
12	5	02 11 13 16 18
13	11	02 07 08 09 11 12 15 16 17 18 19
14	12	01 02 03 04 05 06 09 10 15 16 17 19
15	15	02 03 04 05 06 07 08 09 10 11 13 14 16 17 19
16	14	04 05 07 08 09 10 11 12 13 14 15 17 18 19
17	12	02 05 06 07 09 10 13 14 15 16 18 19
18	7	07 10 12 13 16 17 19
19	12	06 07 08 09 10 11 13 14 15 16 17 18

routing protocol. MeshNet is an indoor 802.11a/b network with 19 nodes and 93 (undirected) links. The original data consists of three ETT datasets which were collected at three different times. The trace files were generated each minute. One trace file contains several lines of text with each line starting with the node's IP address and is followed by pairs of IP addresses and measured ETT values for each neighbor node. To compute the ETT values, two types of probes were used. The first type – broadcast probe – is used to estimate the delivery rate. Every second, each node sends a 524-byte broadcast packet. Neighbor nodes record the number of received packets within 10 seconds. The second type – unicast probe – is used to estimate data bandwidth. Every tenth second, each node sends a unicast pair of packets of size 134 bytes and 1134 bytes to each neighbor. The difference in transmission time of the two packets is piggybacked from neighbors to the node. Using the information gathered by nodes with the above probes, a central server calculated $ETT = packetsize/(d1 \times d2 \times bandwidth)$, where $d1$ and $d2$ are the delivery ratios in the two directions on a link.

We took the ETT values directly from the trace data, processed those ETT values and explored the extracted information to develop a model that can be used to simulate the ETT value. All the results shown in this paper are only for the first dataset because analysis of the other two datasets yielded similar results (but correspond to different topologies as the connectivity appears to have changed). In addition, in the original trace files, the ETT values were missing for some short periods of time, so we use only the largest available continuous trace period (about 350 minutes). Figure 1 depicts the topology of the network at the time the first dataset was collected. Table 1 shows a list of neighbors for each node. We observe that each node has a large node degree (for example, node 15 has 15 neighbors and node 4 has 8 neighbors) indicating the need to characterize several links from each node.

3. BACKGROUND AND RELATED WORK

We were inspired by [5] in which the authors used eigenvector analysis to classify access points according to the number of connected users. This method is also known as Principal Component Analysis (PCA) [6]. In[7], PCA was applied to the received signal strength (RSS) in sensor networks with the goal of reducing its variability with distance and having better prediction of the RSS values. However, to the best of our knowledge, this is the first attempt to model ETT and the first use of PCA towards this. We use PCA to reduce the dimensions of data to extract the most important features from the ETT trace. PCA works as follows. Assume we have an $m \times n$ matrix $\mathbf{X}$. We can reduce the dimensions of X with a small loss in information as follows:

1. Determine the zero-mean $m \times n$ matrix $\mathbf{D} = \mathbf{X} - \bar{\mathbf{X}}$, where $\bar{\mathbf{X}}$ is an $m \times n$ matrix with m repeated rows (a row vector of n values, which are the average of the n columns in $\mathbf{X}$, is repeated). Calculate the $n \times n$ covariance matrix $\mathbf{C}$ of $\mathbf{D}$.

2. Calculate the $n \times 1$ vector of eigenvalues $\mathbf{V}$ and the eigenvectors of $\mathbf{C}$. Denote $\mathbf{F}$ as the matrix, the i-th column of which is the eigenvector corresponding to eigenvalue $\mathbf{V}[i]$. Also assume that $\mathbf{V}$ is sorted in descending order.

3. Choose $k \leq n$ – the number of eigenvalues used to approximate $\mathbf{X}$. Let the $k \times 1$ vector $\mathbf{U}$ be the k selected eigenvalues and and $\mathbf{G}$ be the $n \times k$ matrix of eigenvectors respectively. In this paper we define two indicators to estimate the amount of information lost when $k < n$: coverage and loss. Coverage α is defined as the cumulative sum of the selected normalized eigenvalues and the loss β is the significance of the last selected eigenvalue i.e.,

$$\alpha = \sum_{u=1}^{k} \mathbf{V}[u] / \sum_{u=1}^{n} \mathbf{V}[u]; \quad \beta = \mathbf{V}[k]/\mathbf{V}[1].$$

4. Compute the $m \times k$ matrix $\mathbf{E} = \mathbf{DG}$. Now, we can approximate the matrix $\mathbf{X}$ as $\mathbf{X'} = \mathbf{E}.\mathbf{G}^T + \bar{\mathbf{X}}$, where $[.]^T$ denotes the matrix transpose operator.

If we can choose k to be significantly smaller than n (while we still have large coverage α and small loss β), then we can efficiently represent the matrix $\mathbf{X}$. For instance, from Step 4, we can express the i-th approximated row of $\mathbf{X}$ as: $\mathbf{X}'_i = \mathbf{E}_i.\mathbf{G}^T + \bar{\mathbf{X}}_i$. In other words, we can look at the i-th row of $\mathbf{X}'$ as a linear combination of ONLY k row vectors contained in $\mathbf{G}^T$ (or k column vectors contained in $\mathbf{G}$) with corresponding coefficients contained in the ith row of $\mathbf{E}$:

$$\mathbf{X}'_i = \sum_{u=1}^{k} \mathbf{E}[i,u] \times \mathbf{G}_u^T + \bar{\mathbf{X}}_i$$

This is our objective with the ETT traces as described next. The k column vectors in $\mathbf{G}$ represent the *basis* vectors that capture the information about all of $\mathbf{X}$.

4. ANALYZING ETT WITH PCA

We note here that there are several ETT traces associated with a given node, for each link that originates at that node. They are functions of time. The question is whether we can represent each of these traces as a linear combination of a small set of common basis traces (a) for each node and (b) if possible for the entire network. Then we may be able to characterize the basis traces and the coefficients statistically and use them to develop models for ETT values. For example, let us suppose there are $n = 14$ links from a node and we have a trace for each link. Suppose the 14 traces can be represented in terms of a single trace ($k = 1$) and 14 scalar coefficients multiplying the single trace to yield each separate trace. Then we need to know only the properties of the single trace and the set of 14 scalar coefficients to characterize the ETT traces. As we describe next, it is not possible to use a single basis trace, but we can use *two* $k = 2$ basis traces per node. Also, it is better to use a per-node characterization than a characterization of the ETT traces for the entire network.

4.1 Analyzing ETT traces at each node

4.1.1 Approach and primary observations

We converted the original trace data described in section 2 into matrices, one matrix for one node. Each *row* of a matrix corresponds to the ETT value of a link as a function of time (which is represented by the columns). Thus, the number of rows of a matrix equals the node degree. We use Node 4 as the example in what follows although all nodes exhibit similar features. Figure 2(a) shows the ETT values in the trace (time is in mins). We notice that there are several sharp peaks in the traces that are short-lived. To eliminate such spikes, we use the time average over a step size of $T = 10$ minutes. Figure 2(b) shows the resulting trace (time is in units of 10 mins). The effect of averaging is that we eliminate the irregular high peaks – we return to this issue in Section 6.

We next approximate the matrix $\mathbf{X}$ (with 8 rows and 35 columns for node 4) using $k = 2$ which results in $\alpha = 98\%$ and $\beta = 2.5\%$, both very reasonable. This means, for a given node v, we can express the i-th row of matrix $\mathbf{X}^v$ as

$$\mathbf{X}_i^v = E_{i,1} \times \mathbf{G}_1^T + E_{i,2} \times \mathbf{G}_2^T + \bar{\mathbf{X}}_i \qquad (1)$$

where $\bar{\mathbf{X}}$ is the average ETT across all links from node v at a given time (we recall here that matrix $\bar{\mathbf{X}}$ contains identical rows, so the index i does not play any role) . Note that *each link* from node v can be expressed as a linear combination of the *basis vectors* $\mathbf{G}_1^T$ and $\mathbf{G}_2^T$. In effect, the i-th link from node v is characterized simply by the tuple $(E_{i,1}, E_{i,2})$ which we will call as the *coefficients of the link*. Further, we use linear regression to express $\mathbf{X}_v^i$ *entirely* in terms of the basis vectors by letting $\bar{\mathbf{X}}_i \approx g_1 \times \mathbf{G}_1^T + g_2 \times \mathbf{G}_2^T$. With this approximation, we have

$$\mathbf{X}_i^v = F_{i,1} \times \mathbf{G}_1^T + F_{i,2} \times \mathbf{G}_2^T \qquad (2)$$

where $F_{i,j} = E_{i,j} + g_j, j = 1, 2$ are the *final coefficients* for link i at node v. For node 4, the basis vectors and final coefficients for all of the links are shown in Figure 3.

In Figures 2(b) and (c) we can respectively see the original (averaged) ETT trace and approximated ETT trace for node 04 *before* applying the approximation for $\bar{\mathbf{X}}$. Figure 2(d) plots the final approximated ETT traces. The difference between the approximations and original data is clearer for links with small peaks, such as 04–15, 04–03. For all links, the Kolmogorov-Smirnov test confirms the hypothesis that approximated ETT and original ETT belong to the same distribution. For other nodes of the network, we have gotten very similar results.

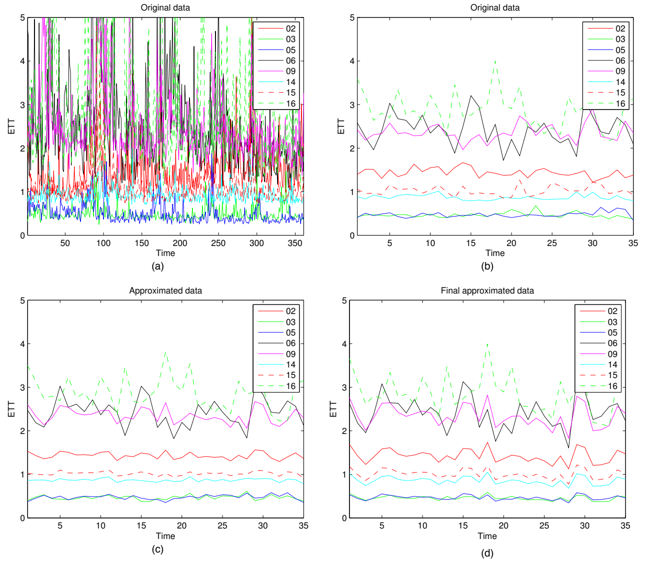

Figure 2: (a) Original ETT trace (b) After averaging for 10 min (c) Approximated ETT values before regression (d) Approximated ETT values after regression

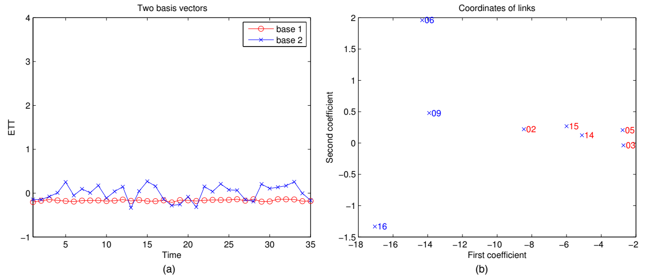

Figure 3: (a) Basis vectors and (b) Final coefficients for Node 4. Links with similar dynamics are highlighted with red color

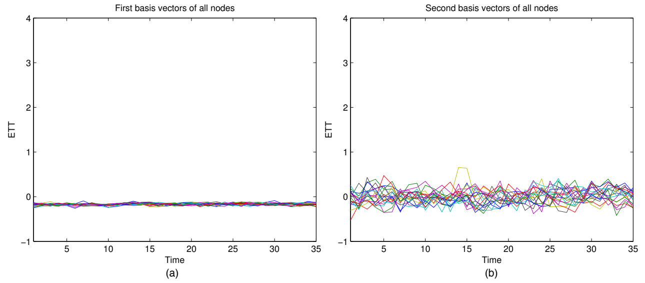

Figure 4: (a) First and (b) second basis vectors from all nodes

Interestingly, we notice that the first basis vector represents the stable component of the ETT traces, while the second one represents the time-varying component. In the coefficients plane links which have similar dynamics form a cluster to the right (shown in red). They also have a small or zero $F_{i,2}$ value. The more fluctuation a link has, the father it is from the horizontal axis. As expected, the two links with high fluctuation 04–06 and 04–16 have largest (in absolute terms) values of the second coefficient.

4.1.2 Analyzing the set of basis vectors of each node

Basic Observations: We already know that each ETT

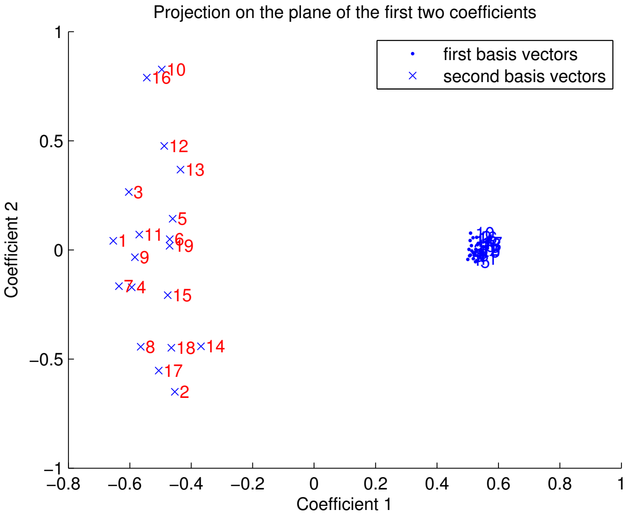

Figure 5: Basis vectors from all nodes

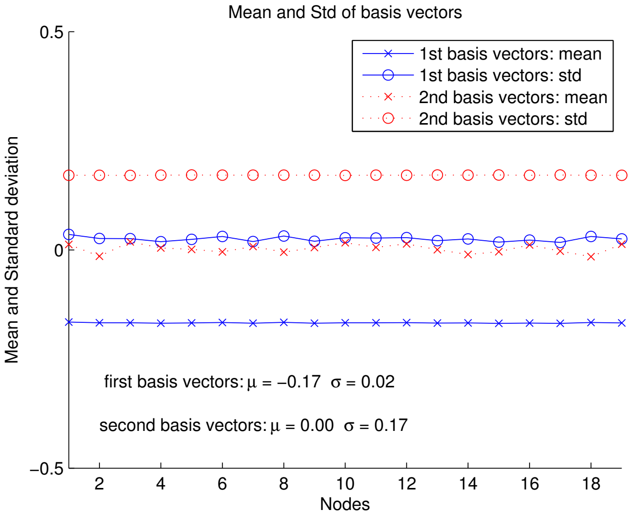

Figure 6: Mean and standard deviation over time of basis vectors of different nodes

trace of links at one node can be approximated using two basis vectors – one represents the stable component, the other represents the fluctuations in time. So what can we say about the relation among basis vectors of different nodes? First, we plot the individual basis vectors from all nodes. Figure 4(a) clearly indicates that all first basis vectors from all nodes are very similar and very stable in time. The second basis vectors, as shown in Figure 4(b), however, are different for each node and also have a larger variation in time. Second, using the same PCA method, we analyzed the matrix built from 19 pairs of basis vectors of all 19 nodes (i.e., a network wide model rather than node by node). To get to a coverage level of $\alpha = 98\%$, we had to use $k = 15$. This means, we need 15 vectors to approximate most of information in 38 basis vectors. In Figure 5 we plotted the first two (over totally 15) coefficients of each approximated pair of basis vectors. As we can see, all the first basis vectors are clustered in one place on the right side, while the second basis vectors are scattered on the left side. This, together with observations from Figure 4, shows that the first basis vectors from all nodes are similar, while the second ones are different.

Statistical Characterization: In Figure 6, we show the mean and standard deviation (over time) of all the basis vectors associated with the 19 nodes. We see that they are the same across nodes in the network leading us to believe that a statistical characterization of the basis vectors may be sufficient to simulate ETT values for any network. We also observe that the set of first basis vectors has a smaller standard deviation (close to zero) showing stability. The set of second basis vectors has near-zero mean but a higher standard deviation. For this preliminary analysis, we assume that (i) the first basis vector is the same for all nodes and constant (ii) the samples of the second basis vectors in time are independent (iii) the second basis vectors across nodes are independent. A plot of the autocorrelation of all and cross-correlation of some pairs of the second basis vector (see Figure 7) indicates that this is reasonable (autocorrelation sidelobes and cross-correlation values are small).

We plotted the two histograms (not shown here) of the first and second basis vectors of all nodes and visually, the values forming the basis vectors have a normal distribution. The individual cumulative distribution functions (CDFs) for each of the first and second basis vectors are shown in Figure 8(a) and (b) respectively. When we employ the averages of the means and standard deviations in Figure 6, as parameters for an expected normal distribution and compare its CDF with the overall CDF built from all of the first and second basis vectors as shown in Figure 8(c) and (d), there is a very good match. Thus, it is possible to generate basis vectors for all nodes simply using network wide values of means and standard deviations for the two basis vectors.

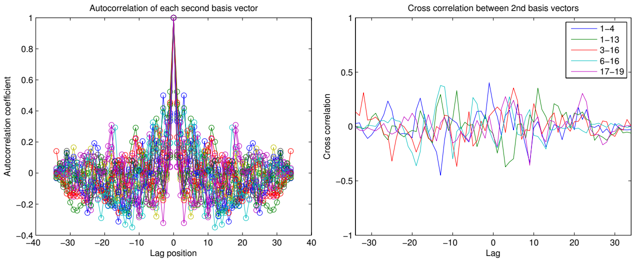

Figure 7: Autocorrelation and Cross–correlation between some of second basis vectors

4.1.3 Analyzing the coefficients of links of each node

Next, we try to characterize the coefficients used to multiply the basis vectors to get the ETT trace. We again consider a per-node analysis of coefficients. Figure 9 shows an overview of all coefficients for all links from all nodes. Most of the coefficients are clustered on the right with $-10 < F_{i,1} < 0$ and $-2 < F_{i,2} < 2$. We do not have any perceivable trends in the nature of these coefficients. As a preliminary step, we assume they are independent from each other and from node to node and link to link and look at their statistics as follows.

Figure 10 shows the normalized histograms for the two coefficients. We see that the second coefficients have a normal-

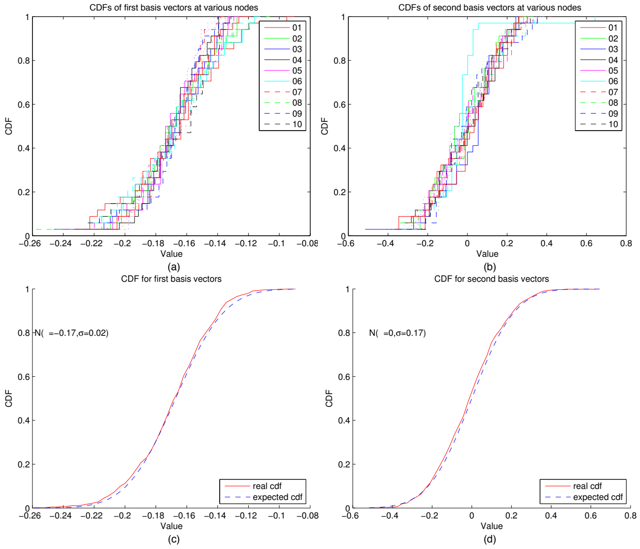

Figure 8: (a) CDFs of first basis vectors from different nodes (b) CDFs of second basis vectors from different nodes (c) Combined CDF of first basis vectors & fit (d) Combined CDF of second basis vectors & fit

like distribution, while the first coefficients have a skewed distribution. Using maximum likelihood estimates, we found that the absolute values of the first coefficients have an *inverse Gaussian distribution* and the second coefficients follow the normal distribution the best. The probability density functions (PDFs) of the estimated distributions are shown on the same plot as the histograms. We note here that the inverse Gaussian distribution has a PDF defined as:

$$pdf(x;\mu,\lambda) = \sqrt{\frac{\lambda}{2\pi x^3}} exp\{\frac{-\lambda(x-\mu)^2}{2\mu^2 x}\} \tag{3}$$

Its mean and variance are μ and μ^3/λ respectively. Once we know mean μ and standard deviation σ, we can calculate the shape parameter $\lambda = \mu^3/\sigma^2$. We use quantile-quantile (Q-Q) plots to test how closely the coefficients fit the respective distributions. We see that the first coefficients fit the inverse Gaussian distribution fairly well. However, the second coefficients do not fit the normal distribution except in the center where most of the coefficients are clustered. There are long tails in the histogram indicating that there is a significant chance of encountering ETT traces with high variability. We recall here that the second coefficient is responsible for the variability in the ETT traces over time.

Although we were able to estimate the distributions of the two coefficients as if they are two independent random variables, clearly, from Figure 9, this is not true. The scatter plot of the two coefficients from all links showed us that they are uncorrelated but dependent random variables. Using k-mean clustering, we found out that coefficients pairs (across all nodes) can be grouped into two clusters (see Figure 12): The first group contains about 70% of the coefficient pairs. While the number of links for some nodes is too small to statistically characterize, based on visual observations (as in Figure 3b, 5 of 8 coefficient pairs form a cluster) we assume that this division can be done on a per node basis as well.

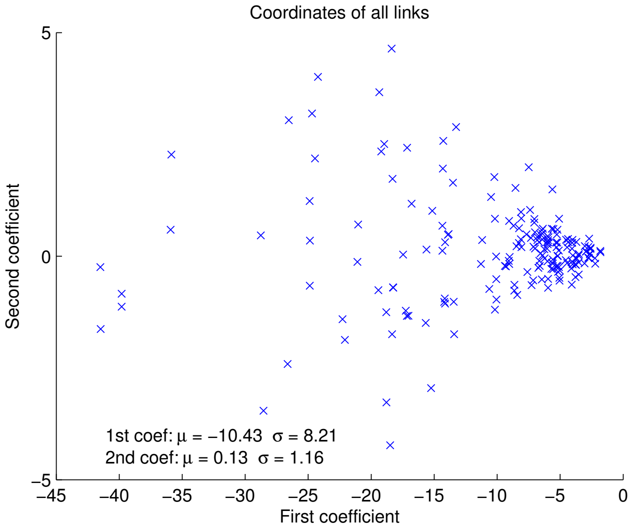

Figure 9: Coefficients of all links

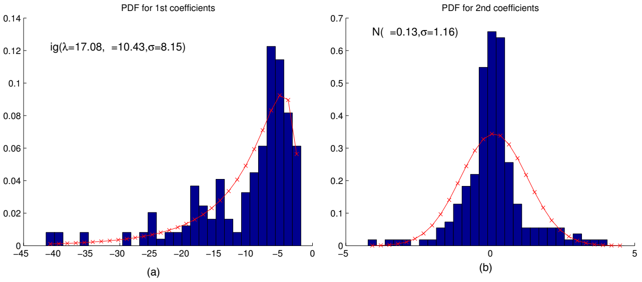

Figure 10: Histogram of (a) all first coefficients (b) all second coefficients

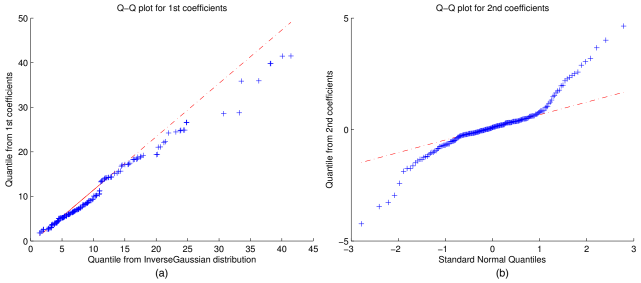

Figure 11: Q–Q plot for (a) first coefficients (b) second coefficients

Figure 13 displays the corresponding silhouette diagram[1]. Comparing the same plot for different data sets, we also observed that the first group occupies a triangular shape

[1] The silhouette diagram plots the silhouette value (ranging from -1 to 1) for each point in each cluster. This value for a point is a measure of how similar that point is to points in its own cluster compared to points in other clusters. The value for point i is calculated as $S(i) = [\min_k b(i,k) - a(i)]/\max\{a(i), \min_k b(i,k)\}$ where $a(i)$ is the average distance from the i-th point to the other points in its cluster, and $b(i,k)$ is the average distance from the i-th point to points in a different cluster k.

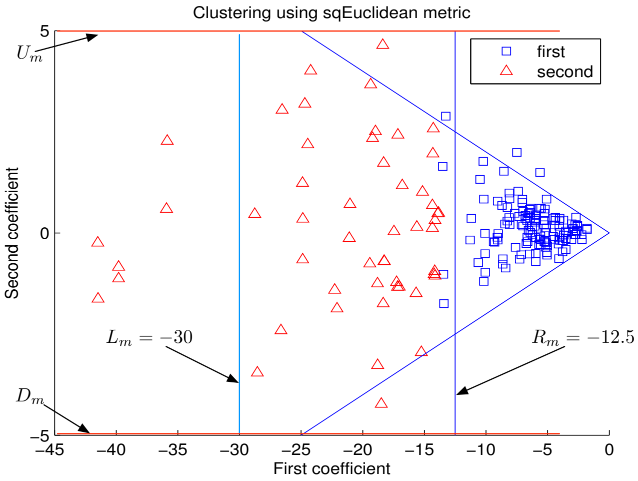

Figure 12: Clustering coefficient pairs

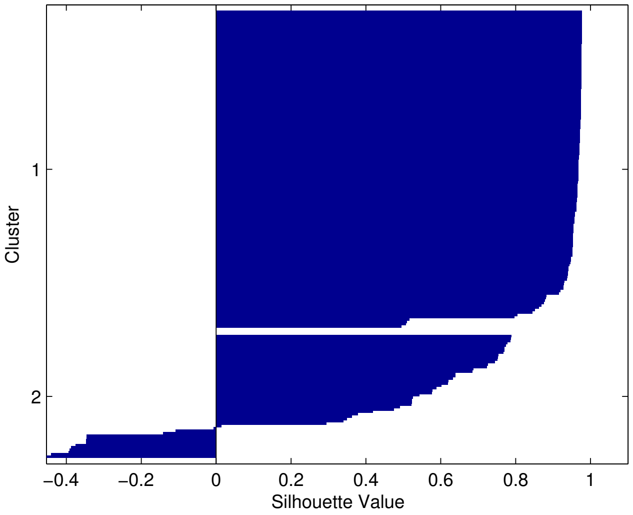

Figure 13: Silhouette diagram

area, as shown in Figure 12. We use this observation in generating the simulation of ETT traces later.

4.1.4 Summary

In summary, we observed that:

- The first basis vectors are stable and considered constant and if necessary be modeled as normally distributed with an almost zero standard deviation. The second basis vectors can be modeled as being normally distributed with zero mean.

- The coefficients that multiply the basis vectors can be grouped into two clusters.

4.2 Analyzing ETT traces for the network

We tried to use the same PCA method to the entire set of ETT traces for all links (instead of only those associated with one node). However, with the same value of $k = 2$, we get only $\alpha = 93\%$ and $\beta = 1.6\%$. This means there are really significant differences in the dynamic behavior of links at different nodes and it is not possible to capture all of these features in a small set of basis vectors.

Table 2: k selection

k	Eigenvalue	Coverage level α	Loss level β
1	0.9150	0.9150	1.0000
2	0.0148	0.9297	0.0161
3	0.0115	0.9413	0.0126
4	0.0083	0.9496	0.0091
5	0.0075	0.9571	0.0082
6	0.0059	0.9630	0.0065
7	0.0056	0.9686	0.0061
8	0.0048	0.9734	0.0053
9	0.0037	0.9771	0.0040
10	0.0031	**0.9801**	0.0033
11	0.0029	0.9830	0.0031

To reach the same level of coverage as we had in the case of the ETT traces from a single node, we have to increase k up to 10 (see Table 2). The resulting traces from an approximation with $k = 10$ still do not pass the statistical tests of fit. However, only when $k \geq 11$ do all of the resulting traces pass the Kolmogorov-Smirnov test.

In Figure 14 we plot the average (over time) error of all links. The vertical lines separate links at different nodes. The node IDs are shown between the vertical lines. Although there are 93 links, in Figure 14 we group links according to the originating nodes. For that reason each link appears as two points in the plot, and we have 186 links (the ETT traces for each link exhibit some differences depending on the direction). We can see that when we increase k from 11 to 20, the average error reduces by almost a factor of 2.With $k = 15$ we have a coverage $\alpha = 99\%$ and only 4 of 93 links have relative approximation errors above 10% (see Figure 15). This means PCA is quite efficient in reducing the amount of data that needs to be stored when applied to link ETTs as a whole. However, the large number of basis vectors and coefficients needed seem to indicate that it is perhaps better to use the per-node analysis to arrive at a general model for simulating ETT values.

5. SIMULATING ETT VALUES

Next we consider employing the previous analysis for simulating ETT data. Recall that we had expressed the estimated ETT on link (v, i) in (2) as:

$$\mathbf{X}_i^v = F_{i,1} \times \mathbf{G}_1^T + F_{i,2} \times \mathbf{G}_2^T \qquad (4)$$

Thus, given a network topology (i.e., the nodes and links that exist) we need to generate a pair of basis vectors $\mathbf{G}_1^T, \mathbf{G}_2^T$ for each node and the scalar coefficients $F_{i,1}, F_{i,2}$ for each link-i from a node. Based on the observations made in Section 4.1.2, we assume that the first basis vector $\mathbf{G}_1^T$ is a constant C. Further, we assume that the second basis vector has samples in time that are independently drawn from a normal distribution with mean 0 and standard deviation σ. We can pick C and σ based on the observations made in Section 4.1.2 (see fig 6), i.e., $C = -0.17$ and $\sigma = 0.17$.

Finding appropriate coefficients to use is more challenging.

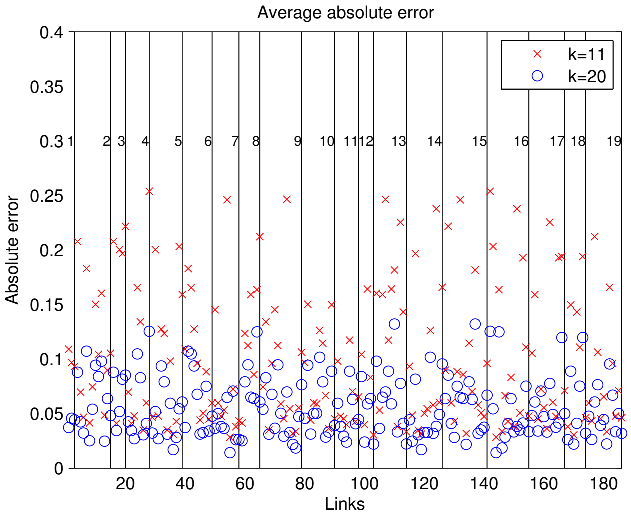

Figure 14: Average error of all links

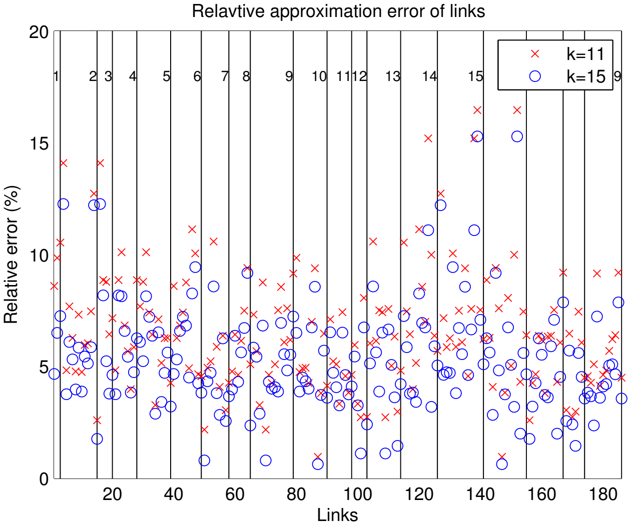

Figure 15: Relative error when $k = 15$

For the sake of simplicity, we suggest the following approach to determine the coefficients:

- Let $\rho = 0.7$ be the fraction of coefficients from the first group in Figure 12; $s = 0.2$ be the slope of the lines that form the triangular region; $L_m = -30, R_m = -12.5$ be the boundary for the first coefficient; $U_m = 5, D_m = -5$ be the boundary for the second coefficient. Generate a uniformly distributed random number x in [0,1] for each link. If $x < \rho$ then we generate coefficients in the triangular region. Otherwise we generate coefficients in the rectangular region as follows.

- Case 1 – triangular region: Generate f_1 uniformly distributed in $(R_m, 0)$. Calculate the range for f_2 as the segment of the vertical line at f_1 truncated by two lines. Let $f_{2D} = s * f_1, f_{2U} = -s * f_1$. Then generate f_2 that is uniformly distributed in (f_{2D}, f_{2U}).

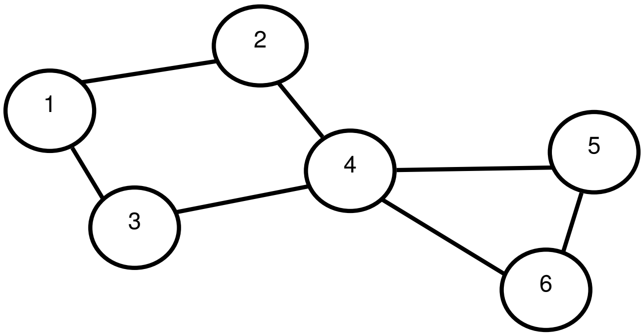

Figure 16: Simple network for simulating ETT values

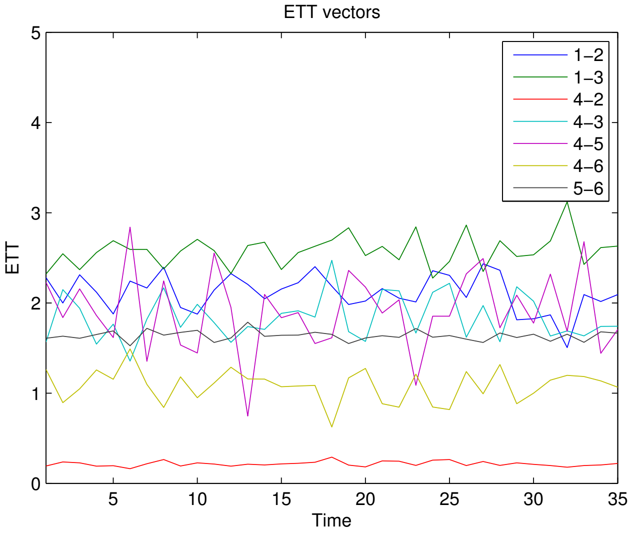

Figure 17: Simulated ETT for a simple network

- Case 2 – rectangular region: Generate f_1 that is uniformly distributed in (L_m, R_m) and f_2 that is uniformly distributed in (D_m, U_m).

We used the above approach to generated simulated ETT traces for a contrived network with six nodes. We used numbers and parameters exactly as described above. The simulated ETT values for this small example network in Figure 16 using this approach are shown in Figure 17. At least visually, they appear to be fairly reasonable.

It is also possible to extend trace data in time (where available) using this approach. Alternatively, we can first generate the coefficients with the available trace data and use these coefficients. We also tried varying the coefficients in the latter case on a rectangular grid around the generated coefficient values and checking the generated traces to see if they statistically match the original trace data with good results.

6. CONCLUSION

In this section, we briefly discuss the limitations of this work and conclusions.

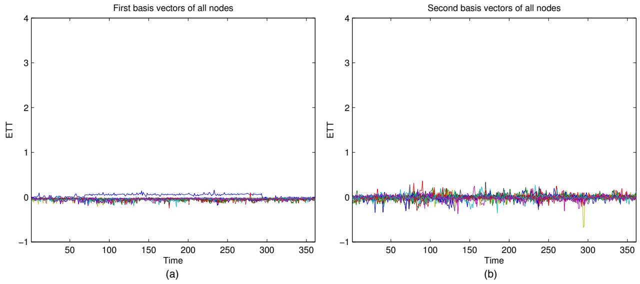

Figure 18: The (a) first and (b) second basis vectors for all nodes if no time averaging of ETT traces is used

6.1 Open Issues

There are some open issues with this work. First, there are several assumptions we have made throughout the paper (e.g., independence, etc.). While such assumptions were used towards developing a simple approach for simulating ETT values, they have not been rigorously tested.

Second, we used time averaged ETT traces to eliminate the spikes as in Figure 2(a). An obvious question is how good this is without the time averaging and whether the spikes matter. We have not analyzed routing protocols or other issues using the original traces and averaged traces. However, we did look at performing PCA on the original traces. We found that it may be possible to still employ only two basis vectors to represent the traces. Figures 18 (a) and (b) show the first and second basis vectors derived for all nodes using the original 350 minute spiky traces. Although the errors between the approximated ETT traces and original traces are higher than without time averaging, it is perhaps possible to employ a similar approach as in this paper. We are looking at this as part of ongoing work.

Third, our results are consistent among the three datasets from the UCSB network that we used. However, we have not tested our approach with trace data from different networks. We are trying to see whether such trace data can be obtained for further analysis.

Fourth, we have not compared our work with alternative approaches for analyzing and modeling the ETT traces. As mentioned in the paper, we were inspired by the work in [5]. However there are other approaches that have been employed for analyzing time varying data. For example, in wireless communications, the time varying nature of the envelope of the received signal (which exhibits Rayleigh fading) has been modeled as a Markov process or a hidden Markov process with two or more states (see for example [8]). The assumption of the future state being dependent only on the current state could be problematic or perhaps not significant (similar to our assumptions of independence).

Fifth, we are also looking at the spatial correlation between ETT values and the possibility of using this approach to predict average ETT values.

6.2 Conclusions

We demonstrated the use of PCA to analyze ETT traces and derive some common features useful for simulation. We have shown that:

- PCA is very useful to reduce the size of ETT trace;
- We can efficiently approximate ETT data of all links at any node using only two basis vectors and two coefficients for each link;
- The first basis vector can be considered as a constant and the second as one derived from a normal distribution with a zero mean;
- The marginal distributions of coefficients corresponding to first basis vectors have an inverse Gaussian distribution, while those corresponding to second basis vectors have a nearly Gaussian distribution;
- It is possible to generate the ETT traces for a given network using our observations or with a combination of existing ETT trace data from that network using only a few parameters.

Acknowledgments

We are grateful to the authors of [4] for making the ETT data publicly available. This work would not have been possible without this data. This research was supported in part by the Army Research Office MURI grant W911NF-07-1-0318.

7. REFERENCES

[1] R. Draves, J. Padhye, and B. Zill, “Routing in multi-radio, multi-hop wireless mesh networks,” in *Proceedings of the 10th annual international conference on Mobile computing and networking*, ser. MobiCom ’04. New York, NY, USA: ACM, 2004, pp. 114–128.

[2] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” in *Proceedings of the 9th annual international conference on Mobile computing and networking*, ser. MobiCom ’03. New York, NY, USA: ACM, 2003, pp. 134–146.

[3] T. Kim, “Cross-layer resilience based on critical points in manets,” Ph.D. dissertation, University of Pittsburgh, 2010. [Online]. Available: http://etd.library.pitt.edu/ETD/available/etd-12172010-120903/

[4] K. Ramachandran, I. Sheriff, E. Belding, and K. Almeroth, “Routing stability in static wireless mesh networks,” in *Proceedings of the 8th international conference on Passive and active network measurement*, ser. PAM’07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 73–83.

[5] F. Calabrese, J. Reades, and C. Ratti, “Eigenplaces: Segmenting space through digital signatures,” *IEEE Pervasive Computing*, vol. 9, pp. 78–84, 2010.

[6] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer, Oct. 2002.

[7] J. Leskovec, P. Sarkar, and C. Guestrin, “Modeling link qualities in a sensor network,” in *Proceedings of the Conference on Data Mining and Data Warehouses*, 2005.

[8] J. Arauz, P. Krishnamurthy, and M. Labrador, “Discrete rayleigh fading channel modeling,” *Wireless Communications and Mobile Computing*, vol. Vol. 4, pp. pp. 413–425, 2004.