NETWORK-WIDE ANOMALY DETECTION BASED ON PCA

Data Mining Final Project Report

Longhai Cui

Ronny Antequera

Mary Sheahen

Weihuan Yang

Abstract:

Computer networks are complex and dynamic since massive generated data need to be transferred constantly among different network domains, for that a network healthy is fundamental since network efficient is required all the time thought network monitoring in order to monitor and diagnose bottlenecks. There are some web based tools like perfSONAR[1] that expose vast data archives of current and historic measurements, which can be queried across end-to-end multi-domain network paths, then automated techniques are required to analyze the measurement data collected on the network in order to detect and notify prominent network anomalies such as plateaus in both real-time and offline manner. The analysis is based on adaptive plateau detection (APD) scheme applied for accurate detection of prominent network anomalies.

The network administrators are more interested in the network level anomalies that have more significant affect to the network rather than path level anomalies. However, most of the anomaly detection today including APD algorithm is only designed to detect path level anomalies. In this project we present a PCA/APD skim that query measurements collected with perfSONAR and analyze network-wide correlated anomalies in measurements of complex dynamic networks with the minimum compute complexity.

1. Introduction:

Given the real time consumption demands huge amount of data is generated and transferred to various places using diverse network technologies, speeds, and domains that can span across continents, for that network performance monitoring is considered required in order to measurement data e.g. one-way delay and TCP throughput captured by monitoring tools like PerfSONAR, considering different algorithms such as APD that will help network administrators to diagnose the bottlenecks in complex networks. In this sense two sort of network anomalies can be detected: uncorrelated and correlated. Uncorrelated network anomaly can be detected at the network-path level by analyzing for e.g., end-to-end one-way delay and throughput measurement time series. Correlated network anomaly events can be detected at the network-wide level by analyzing several network-path level (uncorrelated) anomaly events in order to localize the change and cause to a particular network segment in a certain time. But analyzing large nertwork-paths measurement in not effective due to the amount of data collected.

Collecting data by extracting high-dimensional measurement data from different paths is feasible through software-defined measurement and performance monitor such as Narada metrics [2].

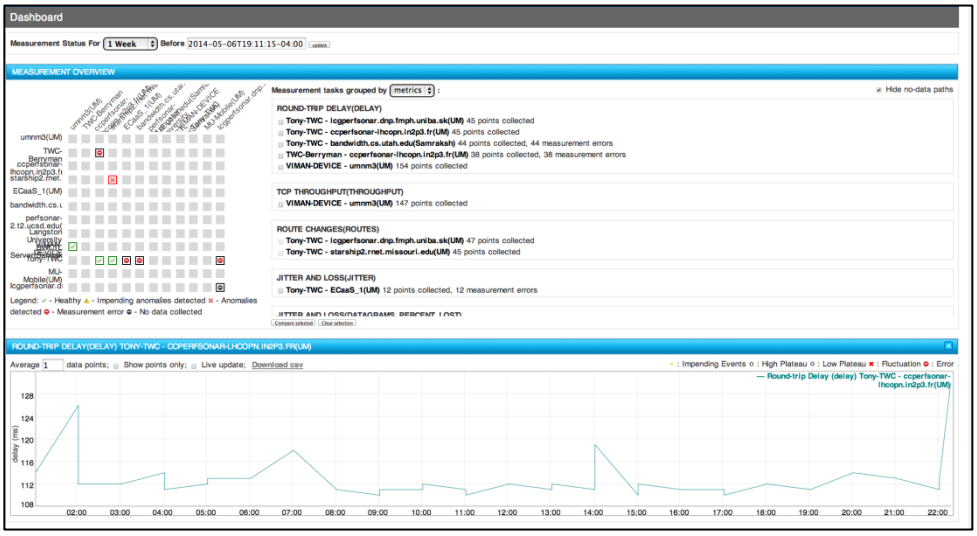

Fig 1. Narada Metrics Dashboard

Once the data is collected we need to apply some detection algorithms that are designed to detect path level anomalies. But usually the there are so many paths in the network and it will cost a lot of time to apply anomaly detection algorithms in all the paths. With the help of PCA algorithm, we can choose the a few number of principle components of the paths that can best represent the overall network performance from the a huge amount of data and still detect all the correlated anomalies. The rest of the report is organized as follows. Section 2 describes our data set. Section 3 describes the methods and workflow of our implementation to detect anomalies. Section 4 shows the results of our experiment and Section 5 conclude the report.

2. Data set:

The first step in our project is to collect data from a network infrastructure, to clearly understand source collected data, we need to differentiate between correlated (all the traces have anomalies at the same time point) and uncorrelated (each trace has anomalies at different time point) anomaly detection, for that the following figures distinguished the kind of anomaly according to the node where is produced.

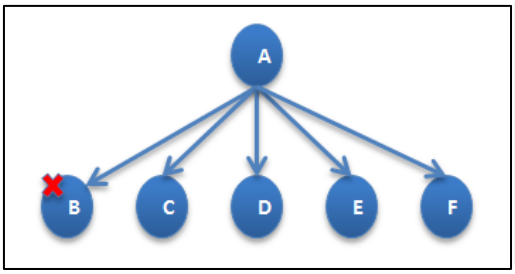

Fig 2. Correlated – Anomaly occurs at the source

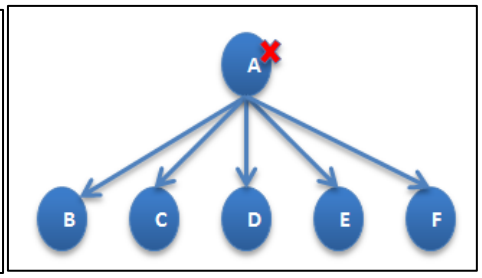

Fig 3. Uncorrelated – Anomaly occurs at the destination

For our project we have collected data one way delay measurement from 15 different paths, each path has 10,000+ measurement data stored in plain text.

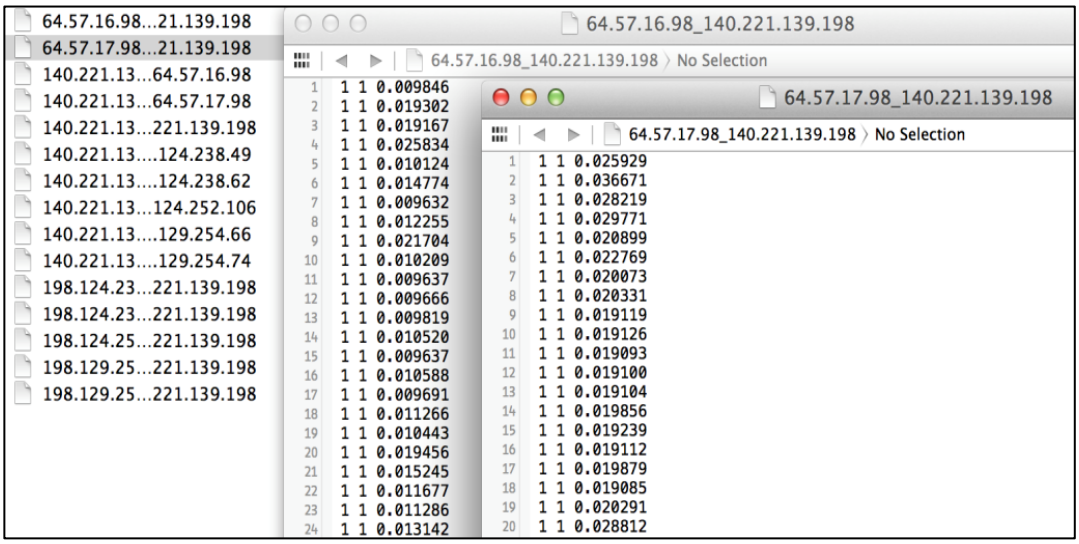

Fig 4. Collected data using measurement monitor application.

From the measurement archive we already know there are 5 Correlated anomaly – all the traces have near time points 3000 4000 5000 6000 7000 and 10 Uncorrelated anomaly – different single traces have anomalies caused by different reasons at different time points. These anomalies will be verified by applying APD and PCA Algorithm presented in the following sections.

3. Approach

3.1 APD Algorithm

The Adaptative Plateau Detection is based on principal components analysis PCA to detect the anomalies in the network-wide level, and show the effectiveness in detecting both correlated and uncorrelated anomalies, since in the algorithm additional parameters “trigger duration” and “sensitivity” are used to update the threshold depending on the curve, resulting in a dynamically changing threshold for anomaly detection then plateaus are detected when the cross a defined threshold and stay there for the defined trigger duration before returning back under the threshold.

The APD algorithm is effective when give a 1-dimensional array for that we must first determine the “network health norm” which is calculated by finding the mean of the data set. In a similar algorithm, this mean allows us to choose a threshold by finding variance from this network health norm then the algorithm finds plateaus by detecting if the values both rise above a given threshold, and stays above that threshold for at least a determined amount of time (trigger duration).

The Adaptive version of this algorithm behaves similarly except for one difference; the threshold is constantly updated, thus making it less likely to have false positives, as seen in the figure below.

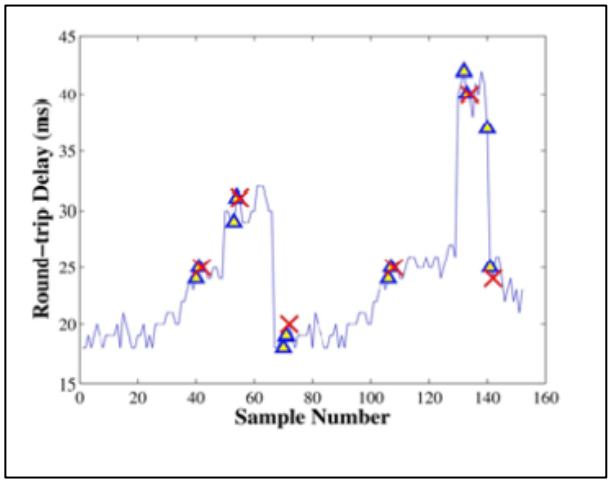

Fig 5. APD algorithm applied

In order to detect the plateaus dynamically, the algorithm uses the variable swc (summary window count). This summary window count is the mean of a subset of the data in a defined window. At each given point in time, we compare the swc of points in the future to the current swc. If points in the future are much higher, the threshold will be lowered proportionately; the opposite is true as well.

3.2 PCA Algorithm

Although this algorithm effectively determines the plateau anomalies in path level arrays, it is not efficient in detected network wide anomalies on its own, so we use the PCA algorithm in order to allow us to do this.

Principle Component Analysis is a coordinate transformation method that maps a given set of data points onto new axes. These axes are called the principal axes or principal components. The principal axes are ordered by the amount of data variance that they capture. The first principal component captures the variance of the data to greatest degree possible on a single axis.

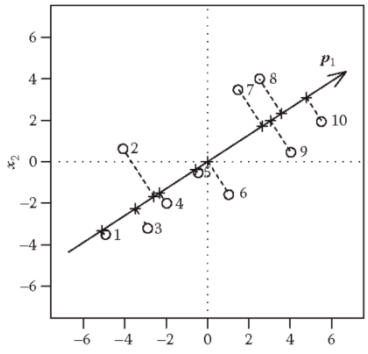

Fig 6. PCA background

3.3 Processing raw data with PCA[3]

Step 1: Read the all the data set and generate a matrix from the measurement data arrays (In our case X=15* 10,000)

Step 2: Calculate V whose columns are the eigenvectors of the covariance matrix of the original matrix and put them in decreasing order.

Step 3: Let the user choose the number of principle components and get reduced matrix R with r dimension.

Step 4: Project the whole matrix to the selected dimensions to get new matrix X_hat.

Step 5: Project the X_hat to one dimension by adding the data in the same columns.

A portion of matlab code is provided bellow to help to understand this data preprocessing steps.

```
C = X'*X; %variance=1/(T-1)*(X*V)'*(X*V)

[V D] = eig(C) %Columns of V are the e-vectors
R = V(:,1:r); %Number of Principal Components

X_hat = R*R'*X'; %Projections to selected dimensions.

X_residual=sum(X_hat,1); %project to one dimension to apply APD
```

3.4 Workflow

The following workflow show us the fused reoriented data comprising of eigen vectors, where the first eigen vector captures maximum variability and the last is left with minimum variability. What this translates into in reality is that the data projection using the first eigen vector has variability that is common to most of datasets and the last eigen vectors have the variability that is least common in the dataset

Fig 7. PCA-with-APD block diagram

Next data dimension selection is performed on the fused reoriented data. For example if we are interested only in the common anomalies we will select only the first eigen vector. After the data dimension (number of eigen vectors) is selected, the data is projected using the eigen vectors and an anomaly detector is used to detect the anomalies.

4.Results:

By applying ADP/PCA we successfully detected 5 Correlated anomalies and 10 uncorrelated anomalies from the data set collected. The following graphs represent the projection of the anomalies in 1, 5 and 15 dimensions, which demonstrate the efficiency of the algorithms in order to find anomalies.

As we can see, we can detect all the 5 network correlated anomalies even with the only 1 principle component. We show that with the number of principle components the users choose growing, we can detect more uncorrelated anomalies in the cost of computation time. The reason is obvious and that is because if we

choose more samples from the original data set, more information we will have about the network. **Note that the uncorrelated anomalies are not our focus since we can just spend a lot of time to run the detection algorithm to each path and get the results, which are even not always helpful for the network administrators. The point of the whole project is to show a skim to detect correlated anomalies with the reduced computation complexity.**

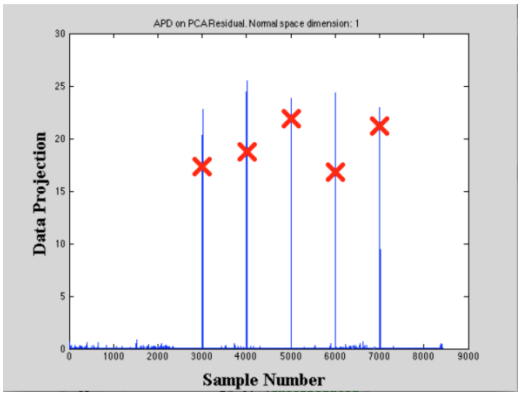

Fig 6. ADP/PCA, Dimension: 1

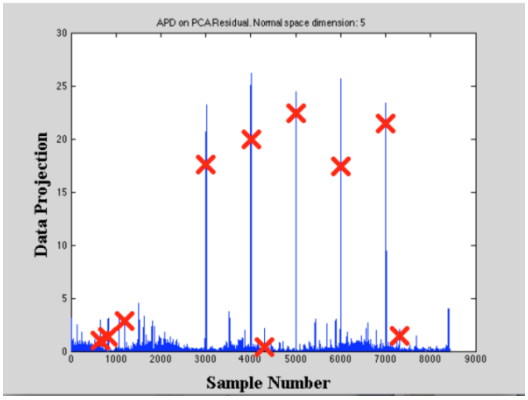

Fig 7. ADP/PCA, Dimension: 5

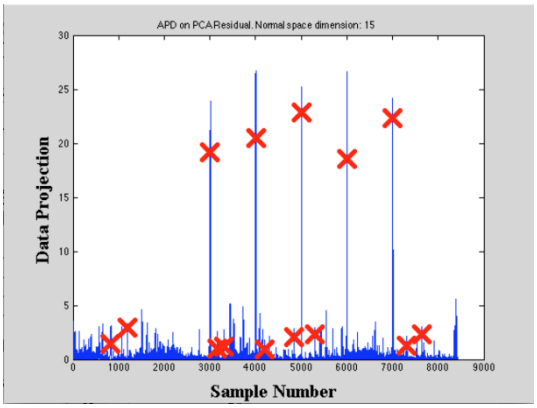

Fig 8. ADP/PCA, Dimension: 15

5. Conclusion:

We presented how we use APD based on PCA principal components analysis to detect and diagnose the network-wide correlated anomalies. Our results show that we can even choose only 1 principal component of our 15 different path level measurement data to get all the 5 correlated anomalies. Given the huge and complex network measurement data, the skim can reduce the computation time significantly and aid to provide useful information of correlated anomalies to the network administrators.

References

[1] Infrastructure for network performance monitoring, http://www.perfsonar.net.

[2] Software-defined measurement and performance monitoring framework, https://www.naradametrics.net.

[3] A.Lakhina,M.Crovella,C.Diot,"DiagnosingNetwork-WideTrafficAnoma- lies", *Proc. of ACM SIGCOMM*, 2004.