

# Extracting Baseline Patterns in Internet Traffic Using Robust Principal Components

Vidarshana W. Bandara and Anura P. Jayasumana

Department of Electrical and Computer Engineering

Colorado State University

Fort Collins, CO 80523-1373, USA

{vwb, anura}@engr.colostate.edu

**Abstract**—Robust BaseLine (RBL) is a formal technique for extracting the baseline of network traffic to capture the underlying traffic trend. A range of applications such as anomaly detection and load balancing rely on baseline estimation. Once the fundamental period of the pattern for analysis is recognized, e.g., based on user interest or a period detector such as Autocorrelation Function (ACF), the basic extraction is carried out in two steps. First, the common component across the dataset is separated using Robust Principal Component Analysis (RPCA). The fundamental pattern in the common component is extracted using Principal Component Analysis (PCA) in the second step. Scaling factors required to fit the base-pattern back into the data are returned automatically by PCA. Two types of traffic baselines may be extracted: RBL-L captures the common behavior across time on a single link, and RBL-N captures the common behavior across a network of links, i.e., in space. RBL-N is particularly useful for specifying traffic matrices more efficiently over time, which normally requires multiple updates to follow baseline trends. The derived base-patterns for a single link or a single time period is then extended over the entire network or thru the entire observation period with a compressive analysis. The compressed base-pattern provides a smoother baseline and also a filter to separate baseline traffic and the deviations on the fly from traffic measurements. When compared against BLGBA (Baseline for Automatic Backbone Management) the proposed scheme provides a less noisy, more precisely fitting baseline. It is also more effective in revealing anomalies.

**Keywords** - Traffic characterization; Baseline; Internet Traffic; Anomaly detection; Load balancing

## I. INTRODUCTION

Trends in traffic such as peaks during busy hours and valleys during inert hours are natural occurrences. Traffic baselines represent such general trends. These baselines, which often are repetitive and perhaps deterministic, carry a large fraction of information about the traffic and play a vital role in traffic engineering, network design, load balancing and pricing. Extracting the baseline behavior from a traffic trace is a subjective task, based on how the baseline is perceived.

The fundamental structure of the baseline is termed “baseline pattern.” The baseline of a traffic trace may be viewed as a series of scaled baseline patterns. The baseline pattern (or base-pattern) captures the following ideas: (1) it represents a segment that cannot be broken down to smaller

similar segments, (2) it is repeated persistently on a trace, and (3) it contains most of the energy of the signal. The base-pattern in effect captures the most prominent features of the traffic trace such as modes, trends and gradients. Having a simple and compact representation for the base-pattern is useful for applications such as characterization of network traffic in terms of traffic matrices, which otherwise would require frequent updates.

### A. Contribution

This paper develops a novel formal scheme for extracting a Robust Base-line (RBL) of a traffic trace. Given a traffic trace, the scheme returns the most common and prominent base-pattern in the dataset, along with the scaling coefficients to construct the baseline. The scheme is developed formalizing features an expert would perceive as constituents of a baseline: a common, prominent and perhaps smooth extraction for the data trace. The novelty of the work lies here. Mathematical tools are applied in order to realize the perceptions of a baseline. The scheme employs Robust Principal Component Analysis (RPCA) [3], a technique that recently has received much interest for separating the common component, i.e., the low rank component, across the dataset. The most salient component comprised of significant principal components in the common component is then extracted using classical PCA. Following that, a compressed representation for the base-pattern is proposed. The compressed representation builds a smoother version of the baseline. Furthermore, compressed representation is also used to build a filter to separate baseline from traffic intensity in real-time.

The formal scheme presented is dataset independent. The scheme operates on a data matrix which could either be multiple time windows on a single link or a data for single time window on multiple links. In the former arrangement, the baseline behavior over a link across time is found, and referred to as RBL-Link (RBL-L). Whilst the latter will deliver the baseline behavior across the network over the considered time window, and referred to as RBL-Network (RBL-N). The scheme is also capable of revealing more subtle patterns that are buried in noisy data segments. The tunable parameters and the optimal settings for each parameter are also discussed.

### B. Related Work

The importance of traffic characterization is emphasized in [7]. However, the widely used random-process based traffic models overlook deterministic baseline behaviors [25]. An

This work was supported in part by AFOSR Contract FA9550-10-C-0090, and ERC program of NSF Award Number 0313747.

extensive survey of traffic identification can be found in [13]. Lack of a proper definition of a baseline has challenged traffic characterization attempts. A simulated network is used in [17] for the baseline characterization in a tactical security architecture. In much of the literature, the baseline behavior of traffic is mostly characterized rather than being extracted. The difference here is that, *characterization* is not constructive, i.e., the returned properties are not sufficient to build a baseline trace, whereas *extraction* filters out the baseline trace from the data trace. A more simplistic approach in [9] uses average and variance to characterize traffic, and uses daily variation to account for dynamics. Such approaches are convenient for implementation. The characterization proposed in [9] is tuned for QoS routing. A statistical approach in [6] uses marginal and multi-variate histograms of traffic features to characterize baseline behavior. Use of Principal Component Analysis (PCA) for classifying baseline and anomalous traffic is discussed in [27]. In [8], entropy based clustering is used on a five-tuple characterization (source and destination; IP and port; and the protocol). This work is further extended in [12] by developing a real time traffic filter. An entropy based profiling scheme for attack detection is presented in [14]. A Hidden Markov Model (HMM) is used in [11], which also presents a good survey emphasizing the need of an effective traffic model. The approach in [18] uses a Gaussian mixture model for baselining network traffic. Some models are driven by the nature of traffic, such as burstiness and self-similarity. An alternative is to use a token bucket scheme to meter bursty traffic traces [15]. A seven-tuple characterization in [10] uses a self-similar model parameterized with the minimum, the maximum and the degree of self-similarity using the Hurst exponent. In a more cross field approach to classify network traffic, Grey Level Co-occurrence Matrices (GLCM) are used in [20]. Here, the idea is to interpret the nature of traffic as the texture of an image. BLGBA proposed in [19] serves as the baseline scheme for GBA tool (Gerenciamento de Backbone Automatizado : Automatic Backbone Management). Two types of baseline sets were used in BLGBA: a set labeled *bl-7* having separate baseline for each day of the week, and a set labeled *bl-3* having a baseline for the week days, one for Saturday and one for Sunday. As an extension, [16] uses BLGBA based baseline and k-means clustering for anomaly detection.

Rest of the paper is as follows. Section II explains the theoretical basis related to separating the base-pattern from a data trace. The derived scheme is applied to real data and the results are shown in Section III. Section IV addresses some applications of the proposed scheme. Section V provides a discussion on the scheme and Section VII concludes the paper.

## II. SCHEME FOR BASE-PATTERN SEPARATION

This section explains the formal scheme used to separate base-patterns from traffic patterns. The base-pattern is expected to capture a significant fraction of the traffic behavior, and stand as a good representation for the traffic trend. Therefore the base-pattern has to be (1) always present in the trace, (2) common to all links, (3) a prominent component in the trace, and preferably (4) has a compact representation. Below we consider two arrangements for data; one extracting baseline behavior over time and the other over space - the network. Finding an optimal period to break time-series is discussed in

Section II.B. RPCA based common component separation is detailed in Section II.C and PCA based salient component extraction in Section II.D. Additionally, compressed analysis on the extracted baseline is addressed in Section II.E and a filter to separate baseline from traffic measurements on the fly is discussed in Section II.F.

### A. Data Arrangement

The scheme extracts the baseline of a traffic trace, arranged in a matrix, referred as  $Y$ . Two arrangements are possible: data traces of multiple links over the same period of time, or data on a single link broken into windows. When data traces of multiple links over an arbitrary period  $N$  is arranged into rows, the scheme returns a base-pattern valid for all the considered links over the period. Due to its validity over the space, it is referred as a “spatial” base-pattern. If  $M$  links are considered, then the data matrix  $Y = \{Y_{mn}\}_{M \times N}$  with links  $m=0 \dots (M-1)$  and sample indices  $n=0 \dots (N-1)$ . The goal behind analyzing time windows on a single link is to identify a base-pattern valid over time for the considered link - therefore is referred as the “temporal” base-pattern. Here, the time window  $N$  is chosen as described in the next section. The choice of the arrangement is application dependant. For example, anomaly detection may be more effective with temporal arrangements; whereas traffic characterization may prefer a spatial arrangement.

### B. The Fundamental Period

To best capture temporal properties of the baseline, time-series has to be broken into cyclostationary periods [28] referred to here as the fundamental period. If the entire time-series is  $T$  samples long, and fundamental period is  $N$ , then time-series is broken into  $M$  windows, where  $M = \lfloor T/N \rfloor$ . Then the time series  $y[t]$  is arranged row-wise on a matrix  $Y = \{Y_{mn}\}_{M \times N}$ ; time window  $m=0 \dots (M-1)$  and sample index  $n=0 \dots (N-1)$  s.t.  $t=m \cdot N + n$  and  $Y_{mn} = y[t]$ . A poorly selected period  $N$  will mis-align and truncate patterns, hampering recognition of the best base-pattern.

While Internet traffic in general exhibit trends that repeat week after week, such a period may not necessarily be obvious or clear in other networks. Corresponding time-series may not have well-defined frequency properties. Therefore, alternative methods have to be employed in identifying the fundamental period ( $N$ ) of the trace. Autocorrelation Function (ACF) can be used to estimate the period by posing the candidate period as the lag of the function [1].

$$R_y[\tau] = \frac{E\{[y[k] - \mu, y[k + \tau] - \mu]\}}{\sigma^2} \quad (1)$$

where  $y[k]$  is the time-series,  $\mu$  is the mean of the series,  $\sigma^2$  is the variance of the series, and  $\tau$  is the lag. Then the optimal estimate for period  $N$  is given by:

$$N = \min_{\tau} \arg \max_{\tau} R_y[\tau] \quad (2)$$

If there happened to be multiple  $\tau$  values that will maximize the ACF, then the least is taken. More efficient cycle detectors can be employed when the search space  $N$  is large.

### C. The Common Component Across Time

The most common component in the dataset is identified using Robust-PCA [3]. Different from the classical PCA,

RPCA breaks a given matrix  $Y$  into a low rank component  $L$  and a sparse component  $S$  as in (3).

$$Y = L + S \quad (3)$$

Such that

$$\arg \min_{L, S} \|L\|_* + \lambda \|S\|_1 \quad (4)$$

where  $\|\cdot\|_*$  is the nuclear-norm (the sum of the singular values),  $\|\cdot\|_1$  is the 1-norm, and typically,  $\lambda = 1/\sqrt{\max(M, N)}$  is chosen. This optimization problem is solved as an Augmented Lagrange Multiplier problem [2] with linear convergence.

The rank deficient component  $L$  carries elements common to all rows (i.e, periods or links). The rank deficiency often is interpreted as follows: the pattern in each row is a linear combinations of a few contributing sources. Since the few sources are common across the matrix, this low rank matrix represents the common component in the data. Traffic on a network on the other hand is the result of a large number of traffic sources. However, there are certain underlying repetitive phenomena, such as work hours and work patterns, when aggregated over a large number of users, result in equivalent logical effect on network traffic.

Our interest is on the low rank  $L$  matrix. Each row of  $L$  is the component in the corresponding row of  $Y$  common with rest of the rows in  $Y$ . Thus  $L$  carries the common component across the dataset. The amount of details of traffic split between the sparse and the low rank components are balanced by  $\lambda$ . Here,  $\lambda$  can be treated as tunable parameter to make the low rank matrix (which is of our interest) detailed or less.

#### D. The Salient Component

Next, the salient patterns in the common component of the dataset are identified. They formulate the base-patterns of the dataset. Here we apply PCA [4] to the  $L$  matrix. In a Singular Value Decomposition (SVD) approach to solve PCA,  $L$  decomposes to:

$$L = U \cdot \Sigma \cdot V^T \quad (5)$$

and the principal components (PCs) of  $L$  projected on the basis of  $U^T$  are given by:

$$U^T \cdot L = \Sigma \cdot V^T = \{L_i\}; i=0 \dots (M-1) \quad (6)$$

Each principal component ( $L_i$ ) can be expressed as

$$L_i = \sigma_i \cdot V_i^T \quad (7)$$

where  $\sigma_i = \Sigma_{ii}$ .

Then the least number of PCs that will satisfactorily capture the information in the trace is selected. They are summed to form the salient component ( $p$ ).

$$p = \{L_i\}; i \in I \quad (8)$$

where  $I$  is an index selected, s.t.,

$$\sum_{i \in I} \|L_i\|_2 \geq \alpha \sum_{\forall i} \|L_i\|_2 \quad (9)$$

where  $\arg \min |I|$ , and  $\alpha < 1$ , but close to 1. The selection criteria for the index set  $I$  is the least number of principal components to represent most of the variance of  $L$ . It is important to note that computing the PCs of the  $L$  matrix need not be done explicitly as a part of singular value shrinkage in RPCA the SVD of  $L$  is computed, and therefore PCs of  $L$  can be directly tapped from the algorithm. In selecting PCs to build the base-pattern the parameter  $\alpha$  determines the amount of energy to be

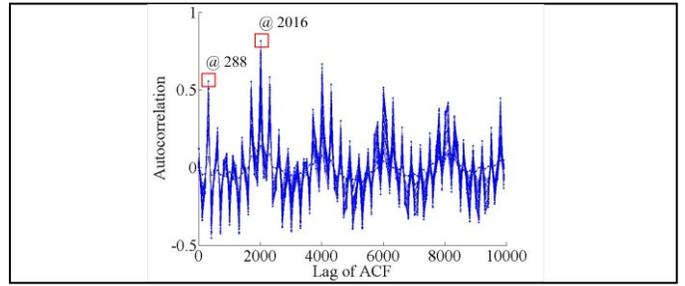


Figure 1. Autocorrelation function at different lags for all links

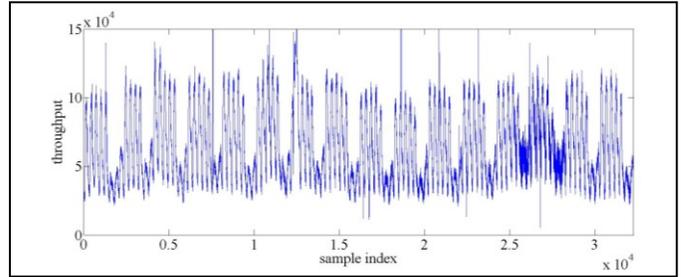


Figure 2. Traffic observed over a period of four months

captured. With a higher  $\alpha$  a base-pattern that resembles the low rank component much closely can be derived. However, that will require selecting more PCs which also calls for maintaining more weights for each realization.

The resulting time-series  $p[n]$  is present in all rows of  $Y$  and captures much of the behavior of the dataset. Thus we call  $p[n]$ ,  $n=0 \dots (N-1)$  as the base-pattern of the dataset  $Y$ . The base pattern can be scaled back into the dataset using the coefficients in  $U$ , constructing a baseline for the dataset. For example the baseline for the  $m^{\text{th}}$  row of  $Y$  is:

$$p_m = \sum_{i \in I} U(m, i) \cdot L_i \quad (10)$$

This baseline is robust against contaminations such as anomalies in the dataset. Thus is referred to as the *Robust Base-Line* (RBL). The RBL constructed with temporal base-patterns yields a robust baseline capturing the baseline behavior of a link; therefore is referred to as RBL-L (RBL for a link). Similarly RBL constructed with spatial base-pattern represent the baseline behavior over the network, and is referred to as RBL-N (RBL for the network).

#### E. Compact Representation

The base-pattern extracted above has a dimensionality of  $N$ . Also the analysis done so far is data independent. It is quite likely that further analysis is possible due to the nature of the Internet traffic data, which will allow a compressed description. Here we propose a compression strategy to reduce the dimensionality of the base-pattern. When long time windows are considered, e.g., weeks-long traces with minutes of sampling intervals, a certain degree of periodicity can be expected, which in turn will allow for a compressed representation of in the Fourier domain. Here, for demonstration purpose the compressibility in the Fourier basis is used. But we acknowledge the possibility of employing other basis appropriate for a given dataset. The dc component of

Fourier domain corresponds to the mean of the time-series. It is automatically included in the base-pattern. Next, to select the other frequency components, consider the Discrete Fourier Transform (DFT) of the base-pattern.

$$P[k] = \sum_{n=0}^{N-1} p[n] \cdot e^{-\frac{2\pi i}{N}nk} \quad (11)$$

We seek a minimum cardinality subset of frequency indices (zeroth frequency is automatically included) that will capture the required energy (variance/information) of the system. Since network traffic traces are real valued data, we also need to maintain the symmetry in the frequency description. Therefore, we select a subset of frequencies  $K$  (referred as the dominant frequency set here forth); s.t.  $\arg \min |K|, k \in K, (N-k) \in K$  and  $k \neq 0$ .

$$\sum_{k \in K} (P[k])^2 \geq \alpha' \sum_{k=1}^{N-1} (P[k])^2 \quad (12)$$

where  $\alpha' < 1$ , and preferable  $> 0.75$ . The resulting set  $K$  of frequency components inclusive of the zeroth frequency is a good smooth approximation to the base-pattern.

#### F. Extending over the Secondary Dimension

To make the compressed description obtained for a single link valid for the entire network or the compressed description obtained for the network over a single time window valid for multiple time windows, the union of the dominant frequency sets is taken. If the dominant frequency set of link- $i$  (or window  $i$ ) is  $K_i$ , and if the network has some  $q$  links (or  $q$  time windows), the dominant frequency set  $K$  for the entire network over a selected time history is,

$$K = K_0 \cup K_1 \cup \dots \cup K_{q-1}. \quad (13)$$

The analysis above permits designing an FFT-filter for base-pattern separation in real-time. The goal of the FFT-filter is to break a given time series ( $y[t]$ ) into its base-pattern component ( $y_B[t]$ ) and the deviation from the base-pattern component ( $y_D[t]$ ) on the fly. The most interesting feature of the filter is, it by-passes all the processing required to obtain the baseline and returns an approximation for the baseline (and the deviation) only with a frequency filtering operation.

$$y[t] = y_B[t] + y_D[t] \quad (14)$$

$$y_B[t] = \frac{1}{N} \sum_{k \in \{0 \cup K\}} e^{\frac{2\pi i}{N}kt} \sum_{j=0}^{N-1} y[j] \cdot e^{-\frac{2\pi i}{N}jk} \quad (15)$$

Similar to (15),  $y_D[t]$  is found by replacing  $\{0 \cup K\}$  with its complement.

### III. RESULTS

In this section we present results of applying the proposed baseline detection for Internet traffic. Results related to compact representation of base-patterns are also included.

#### A. Dataset

The nature of the dataset plays a key role over the extracted baseline. The baseline extracts the prominent and common structures across the dataset. Ideally, it should avoid sudden impulses which may arise due to unusual customer behaviors, attacks, network failures and other causes of traffic anomalies. If the basic structure of traffic - where the peaks and valleys lie, is common, irrespective of increasing, decreasing or even oscillating traffic trends, a common baseline is extracted.

However, if either the different links or the different fundamental periods have disparate traffic trends and structure, there would be a few salient traces characterizing these different groups.

The publically available dataset from Abilene network [5] is used in this work. It consists of 28 links spanning the United States. The dataset provides throughput on the links sampled at five minute intervals, starting from October 16<sup>th</sup>, 2005. At sampling period of five minutes, each link produces 2016 samples a week. For the results presented below, 50 weeks of data on the 28 links are used.

The Abilene dataset however is fairly stable and has similar traffic structures across time. The traffic structure between links is also quite similar. Such a nature results in a single or perhaps a very few salient components. As discussed above this may not be the case with any general traffic dataset. If a dataset of heterogeneous or disparate traffic structures were used, we expect to see more salient components as explained in (8). These components will capture the principal structures of different groups. The weights  $U(m,i)$  in (10) will express how each sample is biased to each salient component. Thus even under heterogeneous traffic trends, the scheme will extract the salient trends as the baseline.

#### B. Estimating the Period

The fundamental period  $N$  of data is found using the ACF as explained Section II.B. Figure 1 shows the variation of ACF for all 28 links over lags up to 10,000. The ACF is maximized at lag 2016, which correspond to a period of a week. Therefore, a week (2016 samples) is selected as the fundamental period  $N$  of the dataset. The ACF for a period of a day (288 samples) is also indicated in the plot. In fact, choosing a week as the duration of the base-pattern can be supported visually as in Fig 2. Figure 2 shows throughput over a period of four months (16 weeks). A pattern that repeats 16 times can be noted in the figure.

#### C. Identifying the Base-Pattern

Data arranged as per Section II.A are decomposed using RPCA to a low rank  $L$  component and a sparse  $S$  component. Figure 3 shows an example of temporal base-pattern extracted for a 10 week long trace. The 10 week data block is broken into separate weeks and arranged as a  $10 \times 2016$  matrix  $Y$  (five rows of which are shown in Fig 3(a)). Then  $Y$  is decomposed to  $L$  and  $S$  (shown in Figs. 3(b) and 3(c) respectively) as per (3) and (4). As expected  $L$  turns out rather similar in all the rows, and differences are pushed to  $S$ . The rank of  $L$  is reduced to 5 from 10 in  $Y$ . Then PCA is applied to the low rank component  $L$  to identify the most salient time-series common to all time windows. A sufficient number of PCs has to be selected to capture much of the details. Figure 4 shows the decay of principal components of raw dataset and of RPCA low rank component, for both spatial and temporal arrangements. The keys in the legend are to be interpreted as, temporal: time-series on a single link, spatial: time-series of multiple links, PCA: PCs of  $Y$ , and base-pattern: PCs of  $L$ . The low rank component of the temporal arrangement has the fastest decaying PCs. In fact the first PC is over an order of magnitude larger than rest of the PCs. Therefore only the first PC is enough to obtain the temporal base-pattern for this dataset.

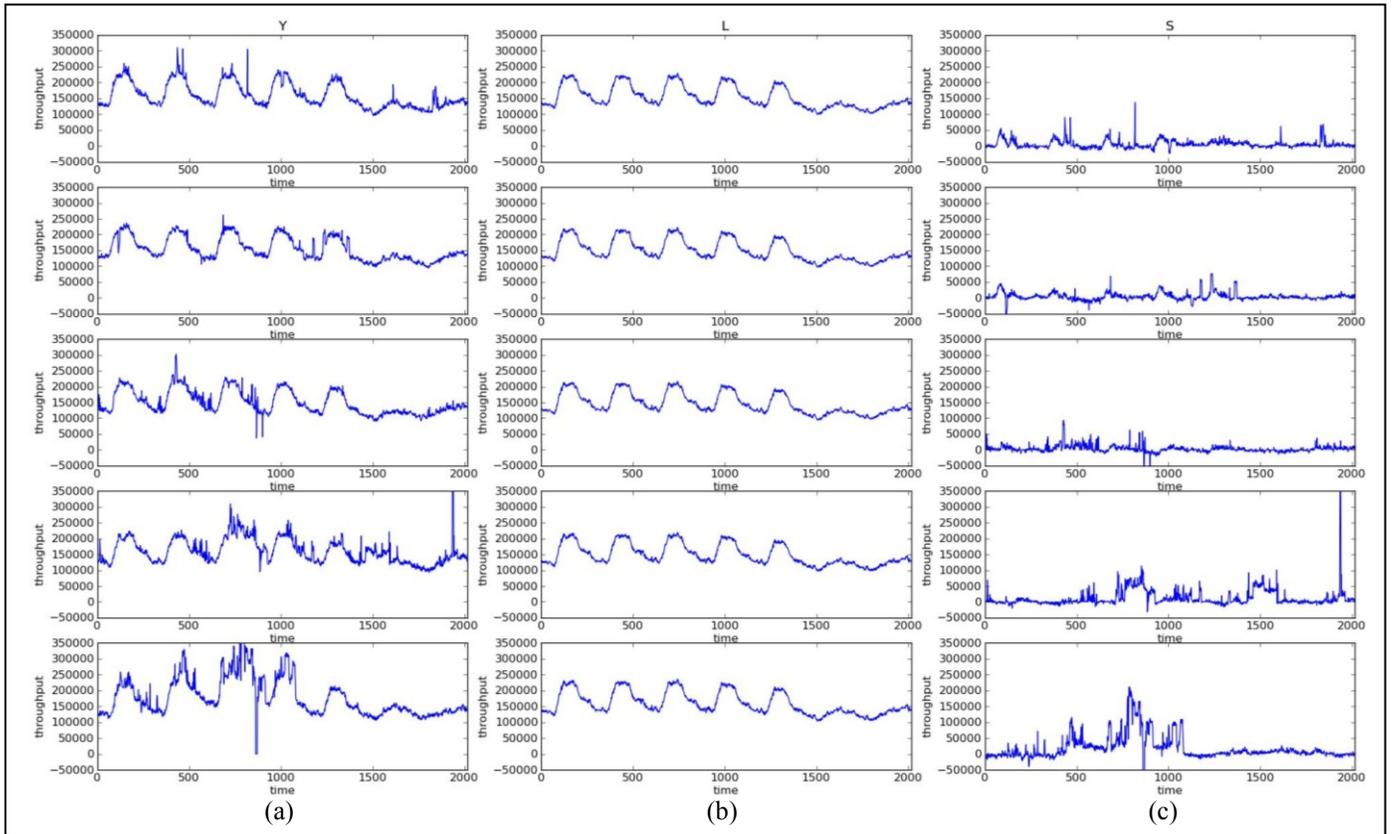


Figure 3. Sparse and low rank decomposition of a 10 week window: (a) data matrix  $Y$ , (b) low rank component  $L$ , and (c) sparse component  $S$  (decomposed throughput vs sample index)

#### D. FFT-Filter

The base-pattern for the Abilene dataset shows certain periodicity suggesting the applicability of the proposed compressed analysis, and the FFT-filter. Fourier analysis revealed the magnitude of the zeroth frequency for any link is at least an order of magnitude greater than other frequency components. After the mean, about 85% of energy on average of all links is concentrated in a subset of 10 frequency components. Thus we build an FFT-filter to select the ten frequency components with largest magnitudes. It was also found that, the superset of these frequencies, even across the entire network consisted of only 13 unique frequencies. This results in a compression from 2016 frequency component base-pattern characterization to a 13 frequency component characterization. Figure 5 shows the filtering performed using the 13 frequency component filter on five sample time series. The resulting filter separates baseline from traffic by-passing the tedious processing required in the scheme. Also, once the FFT-filter is designed, it may be valid over a long period of time due to the stability of the base-pattern as discussed later.

#### E. Comparison

To evaluate the effectiveness of the proposed scheme, we observe how closely the baselines capture raw traffic traces and avoid anomalies. Here we compare RBL-L and RBL-N with temporal and spatial PCA based baselinings [27], bl-7 baseline in BLGBA proposed in [19] and subsequently used in [16][21],

FFT-filter output baseline described in Section II.F, and with low-pass filter baselining, where traffic is filtered thru a low-pass filter to extract the baseline. In BLGBA bl-7 each day of the week is modeled individually. Realizations for each sample is classified by BLGBA into five classes and the maximum value in the class greater than or equal to the 80% percentile is taken to be the baseline. Owing to the nature of the scheme, BLGBA baseline lies above the traffic trace most of the time, i.e., it over-estimates, whereas RBL follows the fundamental trace in the data. BLGBA independently looks at realizations of individual sample points and selects one; therefore it loses the relationship between adjacent samples. This leads to a time-wise uncorrelated noisy base-line. In RBL, relationships between samples are maintained.

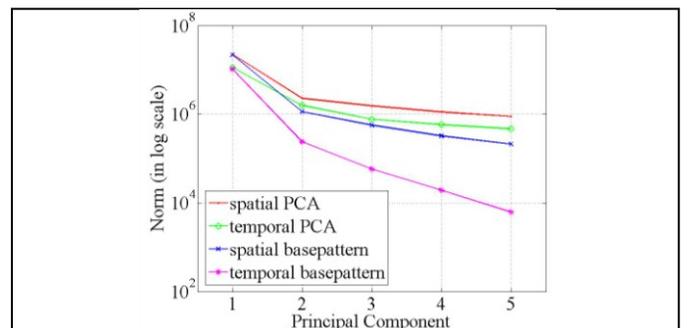


Figure 4. Decay of magnitude of principal components

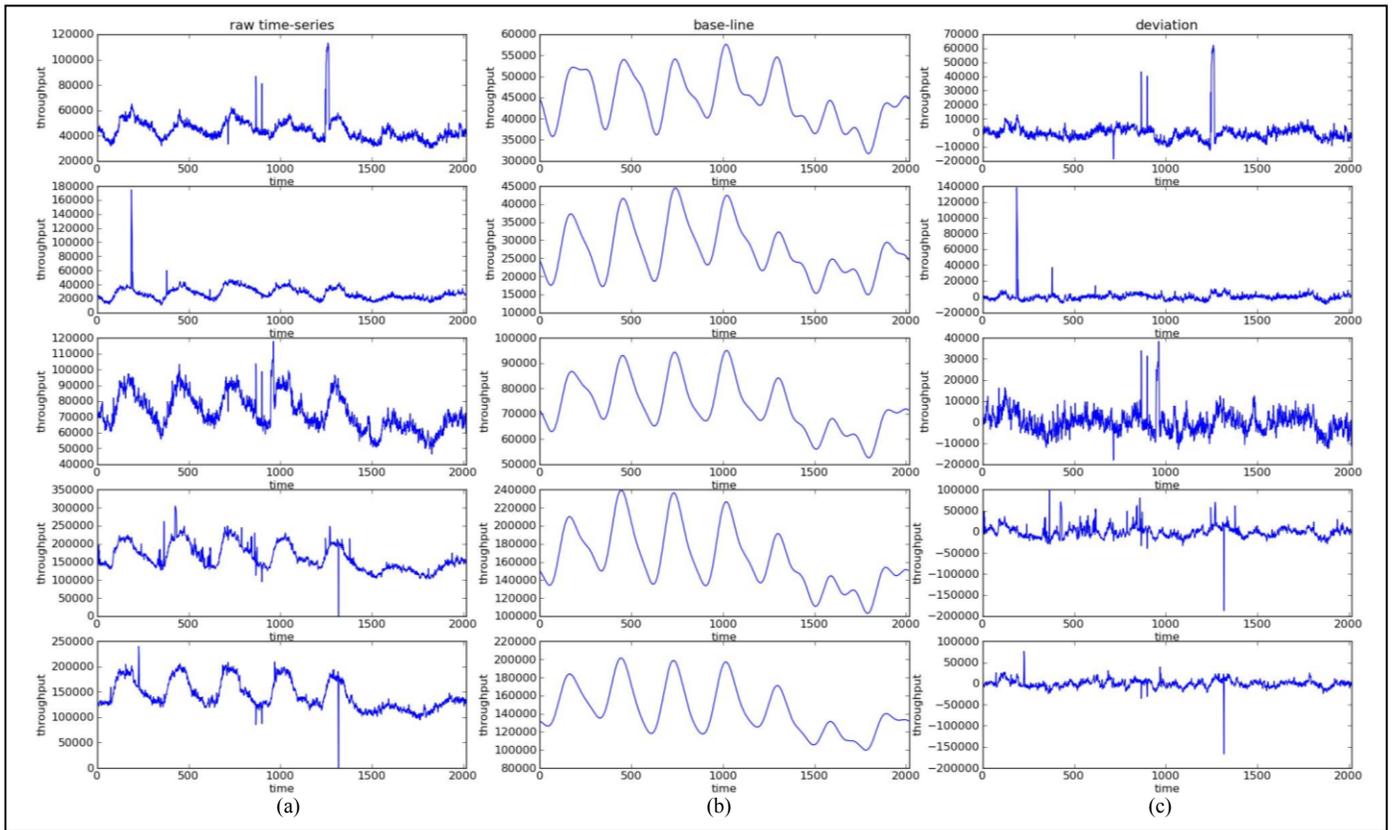


Figure 5. Sample operation of FFT-filter: (a) raw time-series, (b) filtered baseline, (c) deviation of from baseline (decomposed throughput vs sample index)

Further, BLGBA returns one fixed pattern for the entire dataset. How that pattern should be scaled and shifted to fit to individual realizations, i.e., individual links or weeks, is not addressed. In RBL, as the final pattern is the output of PCA, the scaling coefficients  $U(m,i)$  are automatically available. As can be seen in Fig 6, BLGBA is rather noisy defeating the very purposes of having a baseline. RBL baselines are consistent and also display less variance. We do acknowledge the simplicity of BLGBA and the ease of implementation. However, with the availability of hardware such as FPGAs and monitoring devices with significant computation and communication capability, the accuracy gains of RBL is not hard to realize.

In Figure 6 we demonstrate anomaly detection in three selected traces, using each of the baselines. The first (Fig 6(a) and 6(b)) is an example of a typical trace. The second (Fig. 6(c) and 6(d)) has a large anomaly. The third trace (Fig. 6(e) and 6(f)) has a long-lasting anomaly. Figures 6(a), 6(c) and 6(e) show the baselines generated by various methods superimposed on traffic, in an attempt to see how closely each baseline follows traffic. Figures 6(b), 6(d) and 6(f) show the anomalies detected using each method. Anomalies are detected by observing large deviations from baselines. Here the goal is recognize the effectiveness of each baselining technique for anomaly detection. Observing Fig 6(a) BLGBA shows a noisy over-estimating baseline.

The frequency based methods provide a smoother baseline and PCA based methods closely follow the traffic. RBL shows

the best fit to the traffic, whilst avoiding anomalies. As pointed out in [26], PCA based methods can sometimes be vulnerable to large anomalies and certain noise conditions. Figure 6(d) shows an example where PCA methods and RBL-N fail to detect a large anomaly. However, the RBL-L was capable of detecting this anomaly. Anomalies with long durations deform the frequency based baselines, as they form large low frequency descriptions. The long anomaly in Fig 6(f) goes undetected by frequency based methods.

The strength in robust principal component based methods is that they are fairly immune to abrupt changes, such as unexpected customer behaviors. But methods based on classical PCA, and methods that consider every sample with equal importance, such as BLGBA, gain a bias from such. As the baseline is expected to capture the typical behavior, such biases are undesirable. Furthermore, when there are trends in traffic, RPCA identifies them quite effectively even if it is hidden. This is discussed in Section III.G.

#### F. Stability

Here, we investigate the stability of the base-pattern. In Fig. 7 two extractions of base-patterns are considered: using 10 week blocks, and using 5 week blocks. Then the cross correlation between base-pattern of each of the blocks against the base-pattern of the initial block is observed to assess the stability. Furthermore, cross correlation of the data for the same time intervals is also presented, to assess degree of similarity. As can be noticed, with the exception of 5-week base-pattern for the weeks from 10 to 15, others show a high correlation to

the initially derived base-pattern. This indicates that the base-pattern of traffic is fairly stable for this dataset, and it requires less realizations to calculate, as few as five weeks. The cross correlations for baselines always lie above cross correlations for raw data - a verification that the baselines capture the common components in data.

### G. Patterns Buried in Noise

The advantage of baseline detection with a formal scheme is that it may reveal underlying patterns that are more subtle and buried in noise. An example of such is previewed in Figure 8. Figure 8(a) shows weekend traffic on five different weeks. It is not much clear whether an underlying pattern exists between Saturday afternoon and Sunday morning (indicated by the box).

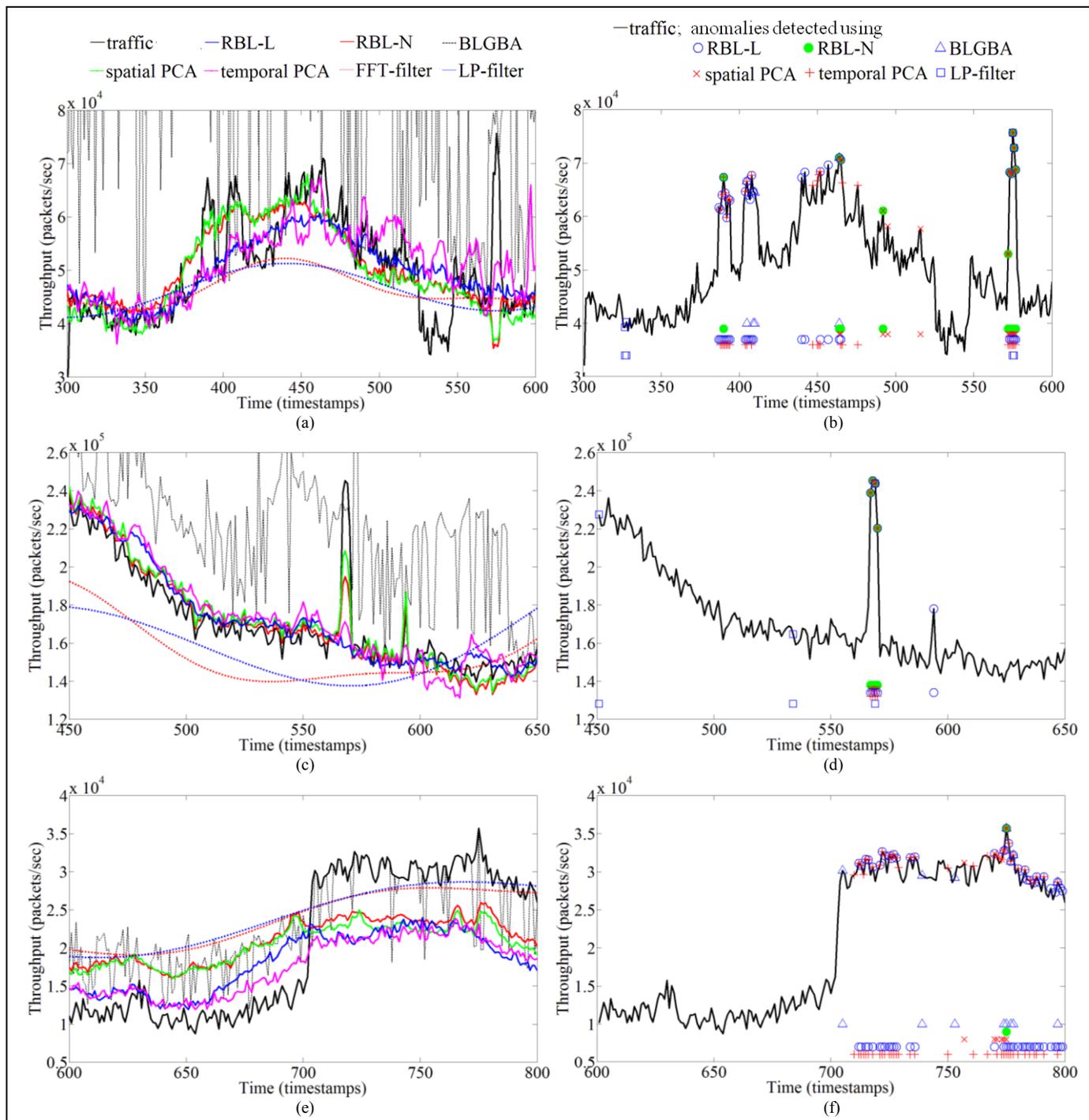


Figure 6. Comparison of baselines and application on anomaly detection: (a) a typical traffic trace, (b) anomalies detected on a typical traffic trace, (c) a trace with a large anomaly, (d) detecting large anomalies, and (e) a trace with a long anomaly (f) detecting a long anomaly (throughput vs sample index). The instances at which anomalies have been detected by different strategies are also identified at the bottom of Figures 6(b),(d),(f), to enhance clarity.

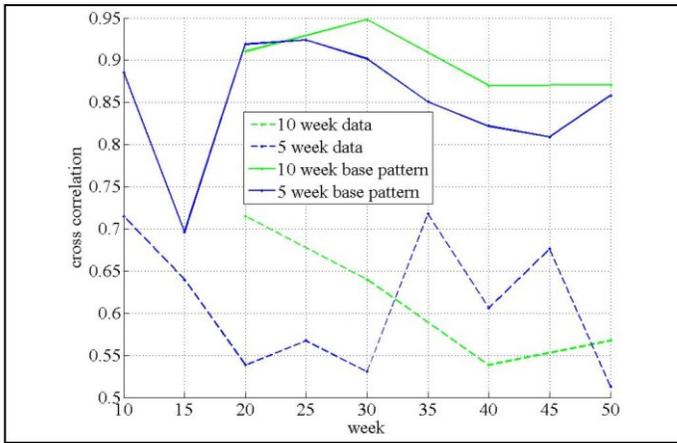


Figure 7. Variation of correlation of the base-pattern and traffic

However, when RBL is applied, it reveals a pattern in Figure 8(b), that can be verified by super-imposing back on the traffic trace as in Figure 8(c).

#### IV. APPLICATIONS

Utilities of the base-pattern  $p[n]$  and its compact version  $y_D[t]$  include a number of applications in traffic engineering. Range of applications includes traffic characterization, load balancing, building robust networks, and pricing. The traffic baseline also may be related to the utilization and the availability of resources required by load balancing

applications. In green computing applications, energy demanding resources can be scheduled to be online based on the demand/utility/base pattern..

Anomalies are large deviations of traffic from the expected. Baselining is an obvious choice for defining the expected traffic. When the difference between the observed traffic is beyond a certain threshold, say, a few times the standard deviation, then the observation is marked as anomalous. With an implementation such as the FFT-filter's deviation component as explained in Section II.F, the deviation from the expected traffic can be readily extracted from the trace. The output here is de-trended for the baseline, and therefore can be thresholded with a fixed value to mark anomalies. Figure 6 also shows few examples demonstrating the effectiveness of using RBL for anomaly detection. Here, all the methods are tuned the same to declare measurements above two standard deviations as anomalous. BLGBA misses a few of the prominent anomalies. PCA based methods overlook large anomalies; while frequency based methods miss long duration anomalies. RBL effectively captures most of the major anomalies in the trace, which escape most other techniques.

Knowledge of baseline traffic behavior is useful in network planning and operations. Features such as resource demand dynamics, maximum and minimum loadings are well described by traffic baselines. Therefore baselines facilitate decisions regarding allocating and scheduling resources. Traffic Matrices (TMs) are a widely used in applications such as QoS routing and network capacity provisioning [24].

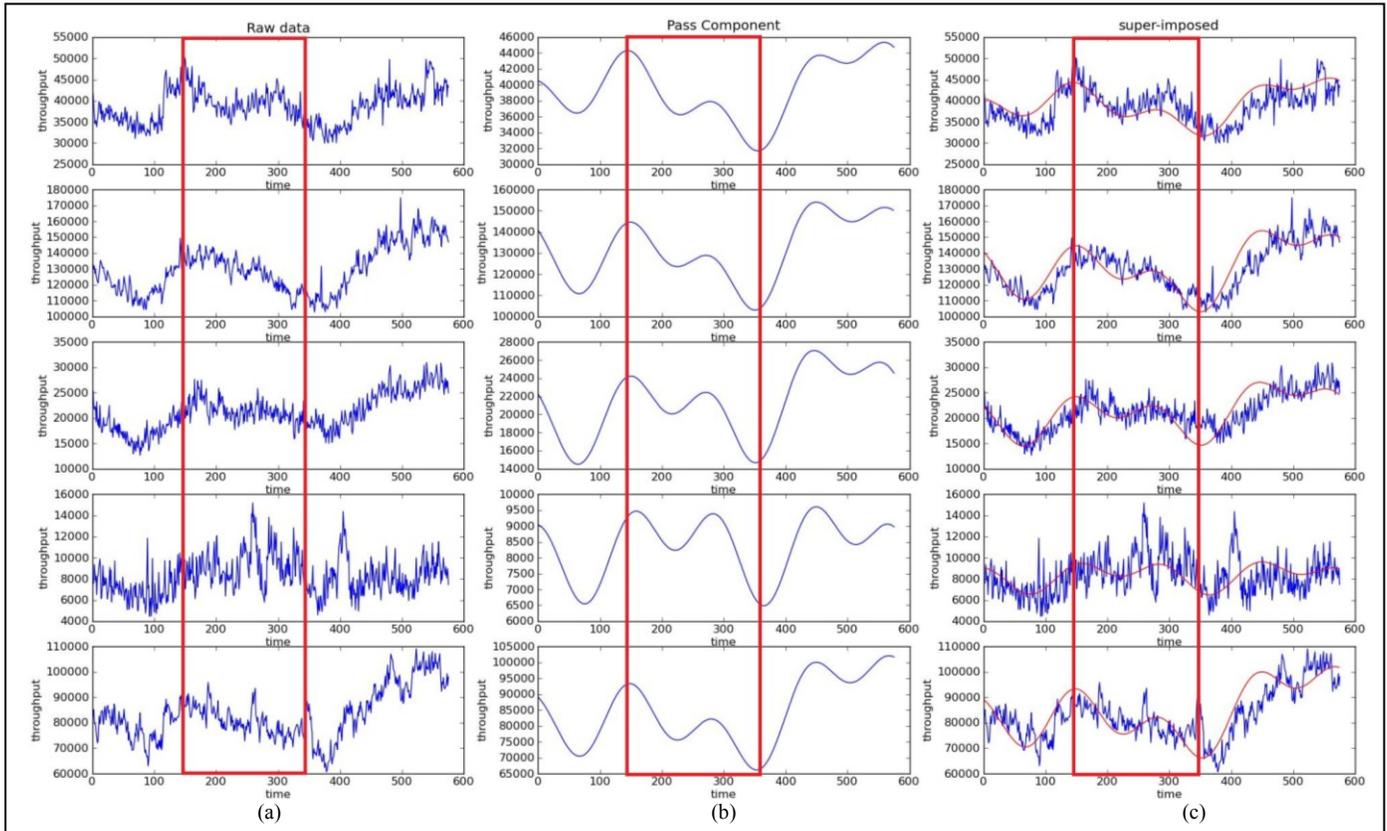


Figure 8. Example revealing subtle patterns; (a) raw time series (b) FFT-filtered smooth baseline; (c) baseline super-imposed on data (decomposed throughput vs sample index)

Methods such as Kalman filtering are often used to capture and predict the dynamics of TMs [23]. By providing the base pattern together with the appropriate scaling factor, TM specification can be simplified. Instead of the frequently updating the entire traffic matrix elements, the few scaling coefficients are needed to be updated only once a fundamental period.

## V. CONCLUSIONS

A scheme for extracting baselines of network traffic traces was presented. The method was illustrated with the spatial and temporal arrangements of traffic to captures common trends in space and time respectively. It consists of two phases. The first uses RPCA to find the most common component of the dataset. Next, PCA was used to extracts the most salient trace in the common component. By identifying common frequency components of the baseline, we constructed an approximate smoother version of the baseline. An FFT filter is then designed to separate such a baseline from a traffic trace in real-time. RBL was found to be more effective in anomaly detection over number of other existing methods.

## ACKNOWLEDGMENT

We thank Dr. Ali Pezeshki, Prof. Louis Scharf, Dr. Randy Paffenroth and Dr. Philip Du Toit for their support extended in this work.

## REFERENCES

- [1] A. Groth, "Estimation of periodicity in time series by ordinal analysis with an application to speech," Preprint series of the DFG SPP 1114, No 14, 2002. B. Fares, P. Apkarian, and D. Noll, "An Augmented Lagrangian Method for a Class of LMI-Constrained Problems in Robust Control Theory," *Internat. J. Control*, vol. 74, 2000, pp. 348-360.
- [2] E. Candes, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis," <http://arxiv.org/abs/0912.3599>, 2009.
- [3] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, August 1987, pp. 37-52.
- [4] <http://abilene.internet2.edu/>
- [5] Y. Kim, J. Jo, K.K. Suh, "Baseline Profile Stability for Network Anomaly Detection," *Proc. of International Conference on Information Technology: New Generations (ITNG)*, April 2006, pp. 720-725.
- [6] K. Claffy, T. Monk, "What's next for Internet data analysis? Status and challenges facing the community," *IEEE Proceedings*, vol. 85, no. 10, Oct 1997, pp. 1563-1571.
- [7] K. Xu, Z. Zhang, S. Bhattacharyya, "Internet Traffic Behavior Profiling for Network Security Monitoring," *IEEE/ACM Transactions on Networking*, vol. 16, no. 6, Dec. 2008, pp. 1241-1252.
- [8] K. Kalapriya, B.R. Raghucharan, A.M. Lele, and S.K. Nandy, "Dynamic traffic profiling for efficient link bandwidth utilization in QoS routing," *Proc. of 2003 Asia-Pacific Conference on Communications (APCC 2003)*, Vol. 2, 21-24 Sept. 2003, pp. 486-493.
- [9] N.M. Garcia, P.P. Monteiro, M.M. Freire, "Measuring and Profiling IP Traffic," *Proc. of 2007 European Conference on Universal Multiservice Networks (ECUMN '07)*, Feb. 2007, pp. 283-291.
- [10] J.E.B. Maia, R.H. Filho, "Internet traffic classification using a Hidden Markov Model," *Proc. of 2010 International Conference on Hybrid Intelligent Systems (HIS)*, 23-25 Aug. 2010, pp. 37-42.
- [11] K. Xu, F. Wang, S. Bhattacharyya, Z. Zhang, "A Real-Time Network Traffic Profiling System," *IEEE/IFIP Proc. of 2007 International Conference on Dependable Systems and Networks (DSN '07)*, 25-28 June 2007, pp. 595-605.
- [12] A. Callado, C. Kamienski, G. Szabo, B. Gero, J. Kelner, S. Fernandes, D. Sadok, "A Survey on Internet Traffic Identification," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 3, 3rd Quarter 2009, pp. 37-52.
- [13] T. Lee, J. He, "Entropy-based profiling of network traffic for detection of security attack," *IEEE Proc. Of 2009 Region 10 Conference (TENCON 2009)*, 23-26 Jan. 2009, pp. 1-5.
- [14] X. Yang, "Designing traffic profiles for bursty Internet traffic," *IEEE Proc. of 2002 Global Telecommunications Conference (GLOBECOM '02)*, vol. 3, 17-21 Nov. 2002, pp. 2149-2154 vol. 3.
- [15] M.F. Lima, B.B. Zarpelão, L.D.H. Sampaio, J.J.P.C. Rodrigues, T. Abrão, M.L. Proença, "Anomaly detection using baseline and K-means clustering," *Software, Proc. of 2010 International Conference on Telecommunications and Computer Networks (SoftCOM)*, 23-25 Sept. 2010, pp. 305-309.
- [16] A.S. Peng, B.R. Eickhoff, T. Hey, D.J. Lilja, "Toward consolidated tactical network architecture: A modeling and simulation study," *IEEE Proc. of 2008 Military Communications Conference (MILCOM 2008)*, 16-19 Nov. 2008, pp. 1-7.
- [17] H. Hajji, "Baselining network traffic and online faults detection," *IEEE Proc. of 2003 International Conference on Communications (ICC '03)*, vol. 1, 11-15 May 2003, pp. 301-308.
- [18] M. L. Proença Jr., C. Coppelmans, M. Botolli, and L. S. Mendes, "Security and reliability in information systems and networks: Baseline to help with network management," *Springer*, 2006, pp. 149-157.
- [19] T. Kisner, A. Essoh, F. Kaderali, "Visualisation of Network Traffic using Dynamic Co-occurrence Matrices," *Proc. of 2007 Internet Monitoring and Protection (ICIMP 2007)*, 1-5 July 2007, pp. 7.
- [20] B. B. Zarpelão, M. L. Proença Jr., L. S. Mendes, "Anomaly detection aiming pro-active management of computer based on digital signature of network segment," *Journal of Network and Systems Management*, v. 15 n. 2, 2007, pp. 267-283.
- [21] G. Terdik, T. Gyires, "Lévy flights and fractal modeling of Internet traffic," *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, Feb. 2009, pp. 120-129.
- [22] L. Yong, C. YuanLi, H. YongXuan, "Kalman filtering based dynamic OD matrix estimation and prediction for traffic systems," *IEEE Proceedings on Intelligent Transportation Systems*, vol. 2, 12-15 Oct. 2003, pp. 1515-1520.
- [23] A. Mansy, C. Dovrolis, "The mythical traffic matrix," 2008, pp. 1-2.
- [24] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and principles of Internet traffic engineering," *RFC 3272*, May 2002.
- [25] H. Ringberg, A. Soule, J. Rexford, and C. Diot, "Sensitivity of PCA for traffic anomaly detection," *Proc. 2007 ACM SIGMETRICS International Conference on Measurement and Modeling of computer systems (SIGMETRICS '07)*. New York, NY, USA, pp. 109-120.
- [26] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E.D. Kolaczyk, and N. Taft, "Structural analysis of network traffic flows," *SIGMETRICS Perform. Eval. Rev.* 32, 1 (June 2004), pp. 61-72.
- [27] W. Gardner, L. Franks, "Characterization of cyclostationary random signal processes," *IEEE Transactions on Information Theory*, vol. 21, no. 1, Jan 1975, pp. 4-14.