# Diagnosing Network-Wide Traffic Anomalies

Anukool Lakhina
Dept. of Computer Science,
Boston University
anukool@cs.bu.edu

Mark Crovella
Dept. of Computer Science,
Boston University
crovella@cs.bu.edu

Christophe Diot*
Intel Research
Cambridge, UK
christophe.diot@intel.com

## ABSTRACT

Anomalies are unusual and significant changes in a network's traffic levels, which can often span multiple links. Diagnosing anomalies is critical for both network operators and end users. It is a difficult problem because one must extract and interpret anomalous patterns from large amounts of high-dimensional, noisy data.

In this paper we propose a general method to diagnose anomalies. This method is based on a separation of the high-dimensional space occupied by a set of network traffic measurements into disjoint subspaces corresponding to normal and anomalous network conditions. We show that this separation can be performed effectively by Principal Component Analysis.

Using only simple traffic measurements from links, we study volume anomalies and show that the method can: (1) accurately detect when a volume anomaly is occurring; (2) correctly identify the underlying origin-destination (OD) flow which is the source of the anomaly; and (3) accurately estimate the amount of traffic involved in the anomalous OD flow.

We evaluate the method's ability to diagnose (*i.e.*, detect, identify, and quantify) both existing and synthetically injected volume anomalies in real traffic from two backbone networks. Our method consistently diagnoses the largest volume anomalies, and does so with a very low false alarm rate.

## Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations

## General Terms

Measurement, Performance, Security

*This work was performed while M. Crovella was at Laboratoire d'Informatique de Paris 6 (LIP6), with support from Centre National de la Recherche Scientifique (CNRS) France and Sprint Labs. Part of this work was done when A. Lakhina was at Intel Research, Cambridge, UK. Data used in the paper was collected while A. Lakhina was at Sprint Labs. This work was supported in part by a grant from Sprint Labs, and by NSF grants ANI-9986397 and CCR-0325701.


*SIGCOMM'04,* Aug. 30–Sept. 3, 2004, Portland, Oregon, USA.


## Keywords

Anomaly Detection, Network Traffic Analysis

## 1. INTRODUCTION

Understanding the nature of traffic anomalies in a network is an important problem. Regardless of whether the anomalies in question are malicious or unintentional, it is important to analyze them for two reasons:

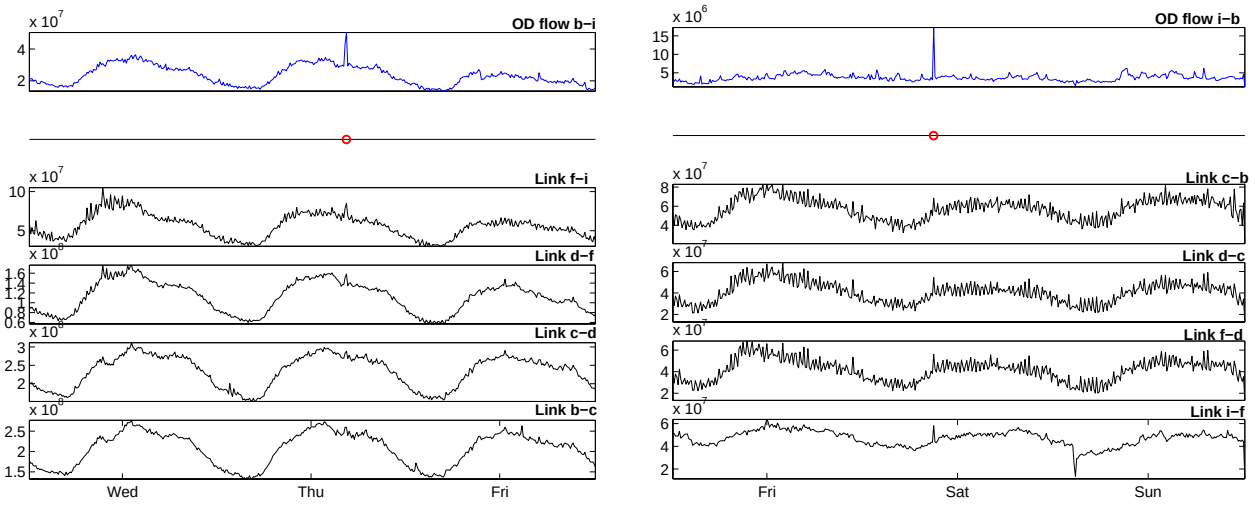
- Anomalies can create congestion in the network and stress resource utilization in a router, which makes them crucial to detect from an operational standpoint.
- Some anomalies may not necessarily impact the network, but they can have a dramatic impact on a customer or the end user.

A significant problem when diagnosing anomalies is that their forms and causes can vary considerably: from Denial of Service (DoS) attacks, to router misconfigurations, to the results of BGP policy modifications.

Despite a large literature on traffic characterization, traffic anomalies remain poorly understood. There are a number of reasons for this. First, identifying anomalies requires a sophisticated monitoring infrastructure. Unfortunately, most ISPs only collect simple traffic measures, *e.g.*, average traffic volumes (using SNMP). More adventurous ISPs do collect flow counts on edge links, but processing the collected data is a demanding task. A second reason for the lack of understanding of traffic anomalies is that ISPs do not have tools for processing measurements that are fast enough to detect anomalies in real time. Thus, ISPs are typically aware of major events (worms or DoS attacks) after the fact, but are generally not able to detect them while they are in progress. A final reason is that the nature of network-wide traffic is high-dimensional and noisy, which makes it difficult to extract meaningful information about anomalies from any kind of traffic statistics.

In this paper we address the problem of diagnosing traffic anomalies that may span multiple links in a network, using link-based statistics. Our approach addresses the anomaly diagnosis problem in three steps: it first uses a general method to detect anomalies in network traffic, then employs distinct methods to identify and quantify them. We believe that this three step approach is appropriate for addressing the large variety of network traffic anomalies. This is because the detection step can target different traffic characteristics such as volume, number of flows, or routing events, while identification and quantification are specific to each type of anomaly.

The goal of this paper is not to explain the cause of network anomalies, but rather to provide a general technique to diagnose




(a) Example 1 (b) Example 2

**Figure 1: Examples of anomalies at the OD flow level (top row) that we want to diagnose from link traffic.**

traffic anomalies. We believe that a necessary first step to discovering the causes of anomalies is the correct detection, identification, and quantification of anomalies.

The contributions of this paper are: (i) a general approach to diagnose anomalies in network traffic, (ii) the application of this method on simple link traffic statistics to isolate "volume anomalies," and (iii) the validation of this method using real data collected on two different backbone networks.

Our method is based on a separation of the space of traffic measurements into normal and anomalous subspaces, by means of Principal Component Analysis. We evaluate the approach using traffic collected from two large backbone networks. We apply the method to both real and synthetically generated anomalies. Our results show that our algorithms are effective at detection (with high detection rate and low false alarm rate); are quite accurate in identifying the underlying OD flow responsible for the anomaly; and are also accurate in estimating the number of bytes in the anomaly.

The paper is organized as follows. In Section 2, we introduce a specific variant of traffic anomalies called volume anomalies and explain why they are important. In Section 3, we describe the data used to illustrate and validate this work. In Section 4, we explain our general approach, which is to decompose the network operating conditions in two subspaces: normal and anomalous. In Section 5, we show how to diagnose volume anomalies by analyzing the anomalous subspace. In Section 6, we validate our approach in two ways: we exploit access to underlying flow data to extract true anomalies using timeseries methods, against which we evaluate detection and false alarm rates and we systematically inject anomalies across time and flows to thoroughly evaluate the method. In Section 7, we discuss further extensions and generalizations of our approach to other types of network anomalies. We contrast our approach from existing work on traffic anomalies in Section 8 and conclude in Section 9.

# 2. VOLUME ANOMALIES

A typical backbone network is composed of nodes (also called Points of Presence, or PoPs) that are connected by links. We define an Origin-Destination (OD) flow as the traffic that enters the backbone at the origin PoP and exits at the destination PoP. The path followed by each OD flow is determined by the routing tables. Therefore, the traffic observed on each backbone link arises from the superposition of these OD flows.

We will use the term *volume anomaly* to refer to a sudden (with respect to timestep used) positive or negative change in an OD flow's traffic. Because such an anomaly originates outside the network, it will propagate from the origin PoP to the destination PoP.

One could detect volume anomalies by collecting IP flow-level traffic summaries on all input links at all PoPs, and applying temporal decomposition methods to each OD flow in the manner of [1, 14]. In general, this is impractical, for a number of reasons. First, there can be hundreds of customer links in a network. Monitoring all input links to collect and aggregate flow level data is extremely resource intensive; for many ISPs this cost is prohibitive. Second, each OD flow would need to be processed separately, requiring estimation of associated parameters for each of the (potentially hundreds of) temporal decompositions.

Instead, we develop a simpler and more practical technique for diagnosing volume anomalies. Given that a volume anomaly propagates through the network, we make use of the fact that we should be able to observe it on all links it traverses. Thus we identify OD flow based anomalies by observing only link counts. (If more detailed information about IP source and destination of an anomaly is then needed, our method can be used as a trigger to indicate which routers need IP flow-level data collection initiated on temporary basis.)

In the next section, we illustrate volume anomalies on link time series taken from a backbone network. Then we propose a diagnosis technique in three steps that relies on the link statistics that are already collected systematically by network operators.

## 2.1 An Illustration

The difficulty of the volume anomaly diagnosis problem stems in part from the fact that it only uses link data (such as can be collected via SNMP). From this link data, one must form inferences about unusual events occurring in the underlying OD flows.

We illustrate this difficulty in Figure 1. The top plot on each side of the figure shows an OD flow timeseries with an associated volume anomaly – this information is *not* available to our algorithms, but we present it to show the nature of the anomalies we are concerned with. The point at which each anomaly occurs is designated

by a circle on the timeline. Below the timeline are plots of link traffic on the four links that carry the given OD flow. These four plots represent the data that is available to our algorithm. Diagnosis of the anomaly consists of processing all link data so as to: (1) correctly detect that at the time shown, the network is experiencing an anomaly; (2) correctly isolate the four links shown as those experiencing the anomaly; and (3) correctly estimate the size of the spike in the OD flow.

We make three observations from these examples. First, while the OD flows have pronounced spikes, the corresponding spike in the link traffic is dwarfed, and difficult to detect even from visual inspection. For instance, the traffic volume at the spike time on links `c-d` and `b-c` in Example 1 is barely distinguishable. Second, the temporal traffic patterns may vary substantially from one link to another. In Example 2, link `i-f` has a smooth trend, whereas the other links for the OD flow have more noisy traffic. Separating the spike from the noise in the traffic on link `c-b` is visually more difficult than separating the spike in link `i-f`. Thus isolating *all* the links exhibiting an anomaly is challenging. Finally, mean traffic levels vary considerably. In Example 1, the mean traffic level on link `c-d` is more than twice that of link `f-i`. The varying traffic levels makes it difficult to estimate the size of the volume anomaly and hence its operational importance.[1]

## 2.2 Problem Definition

The problem of diagnosing a volume anomaly in an OD flow can be separated into three steps: detection, identification and quantification.

The *detection* problem consists of designating those points in time at which the network is experiencing an anomaly. An effective algorithm for solving the detection problem should have a high detection probability and a low false alarm probability.

The *identification* problem consists of selecting the true anomaly type from a set of possible candidate anomalies. The method we propose is extensible to a wide variety of anomalies. However, as a first step in this paper, our candidate anomaly set is the set of all individual OD flows.

Finally, *quantification* is the problem of estimating the number of additional or missing bytes in the underlying traffic flows. Quantification is important because it gives a measure of the importance of the anomaly.

For a successful diagnosis of a volume anomaly, one must be able to detect the time of the anomaly, identify the underlying responsible OD flow, and quantify the size of the anomaly within that flow.

# 3. DATA

Our technique operates on link traffic data, of the kind obtained by SNMP. For validation purposes we also use *OD flow* data, but this data is not an input to our algorithms. All these data have been collected from two backbone networks, Sprint-Europe and Abilene. Note that our anomaly diagnosis method is not limited to backbone networks; it can be applied in any network where link counts are available.

Sprint-Europe (henceforth Sprint) is the European backbone of a US tier-1 ISP. This network has 13 PoPs and carries commercial traffic for large customers (companies, local ISPs, etc.). Abilene is the Internet2 backbone network. It has 11 PoPs and spans the continental USA. Traffic on Abilene is non-commercial, arising mainly from major universities in the US.

[1]Both these examples were successfully diagnosed by the methods in this paper.

| | # PoPs | # Links | Time Bin | Period |
|---|---|---|---|---|
| Sprint-1 | 13 | 49 | 10 min | Jul 07-Jul 13 |
| Sprint-2 | 13 | 49 | 10 min | Aug 11-Aug 17 |
| Abilene | 11 | 41 | 10 min | Apr 07-Apr 13 |

**Table 1: Summary of datasets studied.**

We collected sampled flow data from each router in both networks. For Sprint, we used Cisco's NetFlow [3] to collect every 250th packet (periodic sampling). Packets are aggregated into flows at the network prefix level, and reported in 5 minute bins. On Abilene sampling is random, capturing 1% of all packets. Sampled packets are then aggregated at the 5-tuple level (IP address and port number for both source and destination, along with protocol type) every minute using Juniper's Traffic Sampling [22]. We found good agreement (within 1%-5% accuracy) between sampled flow byte counts, adjusted for sampling rate, and the corresponding SNMP byte counts on links with utilization more than 1 Mbps. Most of the links from both networks fall in this category, and so our sampled flow byte counts are likely to be reasonably accurate. We aggregated both the Sprint and Abilene flow traffic counts into bins of 10 minutes to avoid synchronization issues that could have arisen in the data collection.

Traffic anomalies can last anywhere from milliseconds to hours. Although our method can be used on data with any time granularity, in this paper we work with data binned on 10 minute intervals. In fact, the most prevalent anomalies in our datasets were those that lasted less than 10 minutes and show up as a pronounced spike at a single point in time, as depicted in Figure 1. Thus the anomalies that we detect are those that are significant when traffic is viewed at 10 minute intervals. In Section 7 we will discuss some implications of the particular timescale we use.

To construct OD flows from the raw flows collected, we identify the ingress and egress PoPs of each flow. The ingress PoP can be identified because we collect flows from *each* ingress link in both networks. For egress PoP resolution, we use BGP and ISIS routing tables as detailed in [8]. For Sprint, we supplemented routing tables with router configuration files to resolve customer IP address spaces. Also, Abilene anonymizes the last 11 bits of the destination IP address. This is not a significant concern because there are few prefixes less than 11 bits in the Abilene routing tables, and we found very little traffic destined to these prefixes. Using this procedure, we collected two weeks of complete OD flow traffic counts from Sprint and one week from Abilene. Table 1 summarizes our datasets.

Our diagnosis method operates on link data, so in order to validate against true OD flows we must obtain a set of link traffic counts consistent with the sampled OD flow data collected. To obtain this, we follow the method of [25] and construct link counts from OD flow counts using a routing table taken from the network in operation.

# 4. SUBSPACE ANALYSIS OF LINK TRAFFIC

Effective diagnosis of anomalies in traffic requires the ability to separate them from normal network-wide traffic. In this section, we show how to use Principal Component Analysis (PCA) to separate normal and anomalous network-wide traffic conditions. We begin by stating the relevant notation. We then introduce the basic ideas behind PCA and apply it to the ensemble of link traffic timeseries. Finally, we show how to use PCA to separate the space of link traf-

fic measurements into useful subspaces, representing normal and anomalous traffic behavior. Our anomaly diagnosis method as described in Section 5 builds on the subspace separation developed here.

## 4.1 Notation

The traffic observed on each network link arises from the superposition of OD flows. The relationship between link traffic and OD flow traffic can be concisely captured in the *routing matrix* $\mathbf{A}$. The matrix $\mathbf{A}$ has size (# links) $\times$ (# OD-flows), where $\mathbf{A}_{ij} = 1$ if OD flow $j$ passes over link $i$, and is zero otherwise. Then the vector of traffic counts on links ($\mathbf{y}$) is related to the vector of traffic counts in OD flows ($\mathbf{x}$) by $\mathbf{y} = \mathbf{A}\mathbf{x}$ [23].

Let $m$ denote the number of links in the network and $t$ denote the number of successive time intervals of interest. We let $\mathbf{Y}$ be the $t \times m$ measurement matrix, which denotes the timeseries of all links. Thus, each column $i$ denotes the timeseries of the $i$-th link and each row $j$ represents an instance of all the links at time $j$. In this paper $t$ is the number of 10 minute bins in a week long timeseries (1008) and $m$ is 41 or 49, depending on the network.

While $\mathbf{Y}$ denotes the set of links measurements over time, we will also frequently work just with $\mathbf{y}$, a vector of measurements from a single timestep. Thus $\mathbf{y}$ is an arbitrary row of $\mathbf{Y}$, transposed to a column vector.

We refer to individual columns of a matrix using a single subscript, so the timeseries of measurements of link $i$ is denoted $\mathbf{Y}_i$. All vectors in this paper are column vectors, unless otherwise noted.

Finally, all vectors and matrices will be displayed in boldface; matrices are denoted by upper case letters and vectors by lower case letters.

## 4.2 PCA

PCA 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. When working with zero-mean data, each principal component has the property that it points in the direction of maximum variance remaining in the data, given the variance already accounted for in the preceding components. As such, the first principal component captures the variance of the data to the greatest degree possible on a single axis. The next principal components then each capture the maximum variance among the remaining orthogonal directions. Thus, the principal axes are ordered by the amount of data variance that they capture.

We will apply PCA on our link data matrix $\mathbf{Y}$, treating each row of $\mathbf{Y}$ as a point in $\mathbb{R}^m$. However, before we can do so it is necessary to adjust $\mathbf{Y}$ so that that its columns have zero mean. This ensures that PCA dimensions capture true variance, and thus avoids skewing results due to differences in mean link utilization. For the rest of this paper, $\mathbf{Y}$ will denote the mean-centered link traffic data.

Applying PCA to $\mathbf{Y}$ yields a set of $m$ principal components, $\{\mathbf{v}_i\}_{i=1}^m$. The first principal component $\mathbf{v}_1$ is the vector that points in the direction of maximum variance in $\mathbf{Y}$:

$$\mathbf{v}_1 = \arg\max_{\|\mathbf{v}\|=1} \|\mathbf{Y}\mathbf{v}\|$$

where $\|\mathbf{Y}\mathbf{v}\|^2$ is proportional to the variance of the data measured along $\mathbf{v}$. Proceeding iteratively, once the first $k-1$ principal components have been determined, the $k$-th principal component corresponds to the maximum variance of the residual. The residual is the difference between the original data and the data mapped onto the first $k-1$ principal axes. Thus, we can write the $k$-th principal component $\mathbf{v}_k$ as:

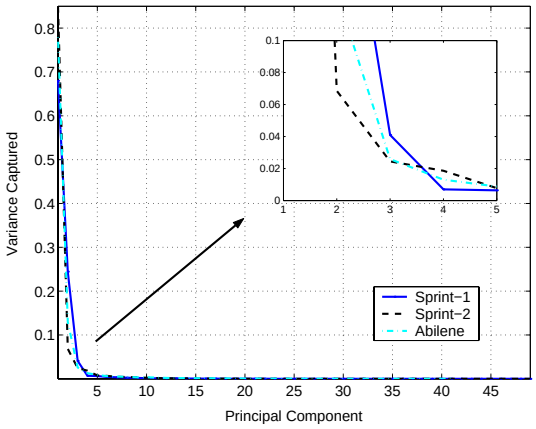



**Figure 2: Fraction of total link traffic variance captured by each principal component.**

$$\mathbf{v}_k = \arg\max_{\|\mathbf{v}\|=1} \|(\mathbf{Y} - \sum_{i=1}^{k-1} \mathbf{Y}\mathbf{v}_i\mathbf{v}_i^T)\mathbf{v}\|.$$

An important use of PCA is to explore the intrinsic dimensionality of a set of data points. By examining the amount of variance captured by each principal component, $\|\mathbf{Y}\mathbf{v}_i\|^2$, we can ask whether most of the variability in the data can be captured in a space of lower dimension. If we find that only the variance along the first $r$ dimensions is non-negligible, then we can conclude that the pointset represented by $\mathbf{Y}$ effectively resides in an $r$-dimensional subspace of $\mathbb{R}^m$.

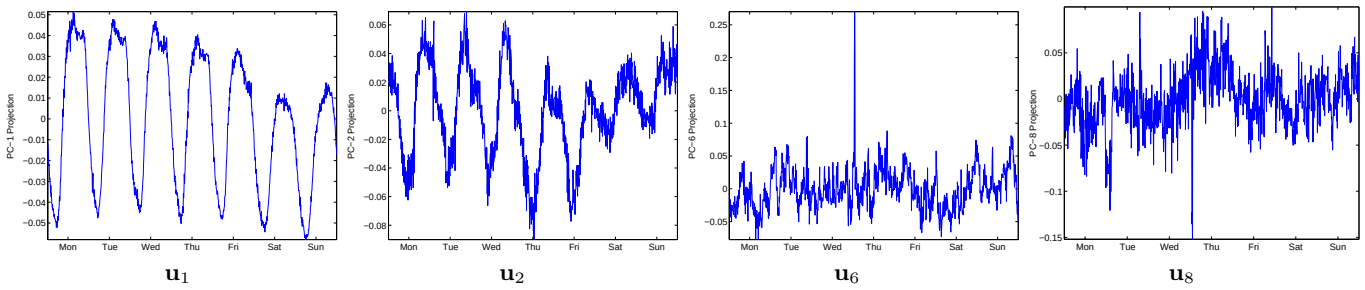
We can observe the phenomenon of low effective dimensionality in our link data. In Figure 2, we plot the fraction of total variance captured by each principal component of $\mathbf{Y}$, for all three of our datasets. This plot reveals that even though both networks have more than 40 links, the vast majority of the variance in each link timeseries can be well captured by 3 or 4 principal components. This low effective dimensionality of link timeseries is consistent with the finding that the underlying OD flows themselves have low intrinisic dimensionality [16]. In fact, the low effective dimensionality of link traffic forms the basis for the success of the subspace methods we describe in the following sections.

## 4.3 Subspace construction via PCA

Once the principal axes have been determined, the dataset can be mapped onto the new axes. The mapping of the data to principal axis $i$ is given by $\mathbf{Y}\mathbf{v}_i$. This vector can be normalized to unit length by dividing it by $\|\mathbf{Y}\mathbf{v}_i\|$. Thus, we have for each principal axis $i$,

$$\mathbf{u}_i = \frac{\mathbf{Y}\mathbf{v}_i}{\|\mathbf{Y}\mathbf{v}_i\|} \quad i = 1, ..., m.$$

The $\mathbf{u}_i$ are vectors of size $t$ and are orthogonal by construction. The above equation shows that all the link counts, when weighted by $\mathbf{v}_i$, produce one dimension of the transformed data. Thus vector $\mathbf{u}_i$ captures the temporal variation common to the entire ensemble of link traffic timeseries along principal axis $i$. Since the principal axes are in order of contribution to overall variance, $\mathbf{u}_1$ captures the strongest temporal trend common to all link traffic, $\mathbf{u}_2$ captures the next strongest, and so on. Specifically as Figure 2 shows, the set $\{\mathbf{u}_i\}_{i=1}^4$ captures most of the variance and hence the most significant temporal patterns common to the ensemble of all link traffic timeseries.




$\mathbf{u}_1$ $\mathbf{u}_2$ $\mathbf{u}_6$ $\mathbf{u}_8$

(a) Normal Behavior (b) Anomalous Behavior

**Figure 3: Projections onto principal components showing normal and anomalous traffic variation.**

The subspace method works by separating the principal axes into two sets, corresponding to *normal* and *anomalous* variation in traffic. The space spanned by the set of normal axes is the normal subspace $\mathcal{S}$ and the space spanned by the anomalous axes is the anomalous subspace $\tilde{\mathcal{S}}$.

Figure 3 illustrates the difference between normal and anomalous traffic variation, as captured in the PCA decomposition. The figure shows sample projections of the Sprint-1 dataset onto selected principal components. On the left, we show projections onto the first two principal components ($\mathbf{u}_1$ and $\mathbf{u}_2$), which capture the most significant variation in the data. These timeseries are periodic and reasonably deterministic, and clearly capture the typical diurnal patterns which are common across traffic on all links. Note that $\mathbf{u}_1$ and $\mathbf{u}_2$ are roughly 180 degrees out of phase, meaning that the two can be used in linear combination to roughly construct of sinusoid of any phase. Thus the extraction of common temporal patterns via PCA does not require the underlying traffic timeseries to have the same periodic phase (*e.g.*, as reflected in traffic in the same timezone). The subspace method assigns these traffic variations to the normal subspace.

We also show projections $\mathbf{u}_6$ and $\mathbf{u}_8$ on the right side of Figure 3. In contrast to $\mathbf{u}_1$ and $\mathbf{u}_2$, these projections of the data exhibit significant anomalous behavior. These traffic "spikes" indicate unusual network conditions, possibly induced by a volume anomaly at the OD flow level. The subspace method treats such projections of the data as belonging to the anomalous subspace.

A variety of procedures can be applied to separate the two types of projections into normal and anomalous sets. Based on examining the differences between typical and atypical projections (left and right sides of Figure 3), we developed a simple threshold-based separation method that we found to work well in practice. Specifically, our separation procedure examines the projection on each principal axis in order; as soon as a projection is found that exceeds the threshold (*e.g.*, contains a $3\sigma$ deviation from the mean), that principal axis and all subsequent axes are assigned to the anomalous subspace. All previous principal axes then are assigned to the normal subspace. This procedure resulted in placing the first four principal components in the normal subspace in each case; as can be seen from Figure 2, this means that all dimensions showing significant variance are assigned to the normal subspace.

Having separated the space of all possible link traffic measurements into the subspaces $\mathcal{S}$ and $\tilde{\mathcal{S}}$, we can then decompose the traffic on each link into its normal and anomalous components. We show how to use this idea to diagnose volume anomalies in the next section.

# 5. DIAGNOSING VOLUME ANOMALIES

The methods we use for detecting and identifying volume anomalies draw from theory developed for subspace-based fault detection in multivariate process control [5, 6, 11]. Our notation in the following subsections follows [5].

## 5.1 Detection

Detecting volume anomalies in link traffic relies on the separation of link traffic $\mathbf{y}$ at any timestep into normal and anomalous components. We will refer to these as the *modeled* and *residual* parts of $\mathbf{y}$.

The key idea in the subspace-based detection step is that, once $\mathcal{S}$ and $\tilde{\mathcal{S}}$ have been constructed, this separation can be effectively performed by forming the *projection* of link traffic onto these two subspaces. That is, we seek to decompose the set of link measurements at a given point in time $\mathbf{y}$:

$$\mathbf{y} = \hat{\mathbf{y}} + \tilde{\mathbf{y}}$$

such that $\hat{\mathbf{y}}$ corresponds to modeled and $\tilde{\mathbf{y}}$ to residual traffic. We form $\hat{\mathbf{y}}$ by projecting $\mathbf{y}$ onto $\mathcal{S}$, and we form $\tilde{\mathbf{y}}$ by projecting $\mathbf{y}$ onto $\tilde{\mathcal{S}}$.

To accomplish this, we arrange the set of principal components corresponding to the normal subspace $(\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_r)$ as columns of a matrix $\mathbf{P}$ of size $m \times r$ where $r$ denotes the number of normal axes (chosen as described in Section 4.3). We can then write $\hat{\mathbf{y}}$ and $\tilde{\mathbf{y}}$ as:

$$\hat{\mathbf{y}} = \mathbf{P}\mathbf{P}^T\mathbf{y} = \mathbf{C}\mathbf{y} \quad \text{and} \quad \tilde{\mathbf{y}} = \left(\mathbf{I} - \mathbf{P}\mathbf{P}^T\right)\mathbf{y} = \tilde{\mathbf{C}}\mathbf{y}$$

where the matrix $\mathbf{C} = \mathbf{P}\mathbf{P}^T$ represents the linear operator that performs projection onto the normal subspace $\mathcal{S}$, and $\tilde{\mathbf{C}}$ likewise projects onto the anomaly subspace $\tilde{\mathcal{S}}$.

Thus, $\hat{\mathbf{y}}$ contains the modeled traffic and $\tilde{\mathbf{y}}$ the residual traffic. In general, the occurrence of a volume anomaly will tend to result in a large change to $\tilde{\mathbf{y}}$.

A useful statistic for detecting abnormal changes in $\tilde{\mathbf{y}}$ is the squared prediction error (SPE):

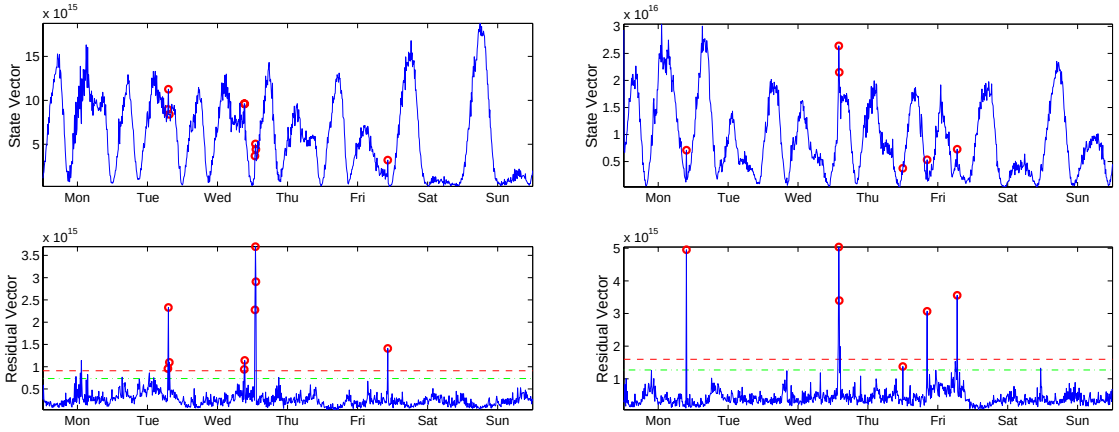
$$\text{SPE} \equiv \|\tilde{\mathbf{y}}\|^2 = \|\tilde{\mathbf{C}}\mathbf{y}\|^2$$

and we may consider network traffic to be normal if

$$\text{SPE} \leq \delta_\alpha^2$$

where $\delta_\alpha^2$ denotes the threshold for the SPE at the $1-\alpha$ confidence level. A statistical test for the residual vector known as the *Q-statistic* was developed by Jackson and Mudholkar and is given in [11] as:

$$\delta_\alpha^2 = \phi_1 \left[ \frac{c_\alpha \sqrt{2\phi_2 h_0^2}}{\phi_1} + 1 + \frac{\phi_2 h_0 (h_0 - 1)}{\phi_1^2} \right]^{\frac{1}{h_0}}$$




(a) Sprint-1 (b) Sprint-2

**Figure 4: Timeseries plots of state vector squared magnitude ($\|\mathbf{y}\|^2$, upper) and residual vector squared magnitude ($\|\tilde{\mathbf{y}}\|^2$, lower) for two weeks of Sprint data.**

where

$$h_0 = 1 - \frac{2\phi_1\phi_3}{3\phi_2^2}, \quad \text{and} \quad \phi_i = \sum_{j=r+1}^{m} \lambda_j^i; \quad \text{for } i = 1, 2, 3$$

and where $\lambda_j$ is the variance captured by projecting the data on the $j$-th principal component ($\|\mathbf{Y}\mathbf{v}_j\|^2$), and $c_\alpha$ is the $1-\alpha$ percentile in a standard normal distribution. Jackson and Mudholkar's result holds regardless of how many principal components are retained in the normal subspace.

Note that in this setting, the $1-\alpha$ confidence limit corresponds to a false alarm rate of $\alpha$, if the assumptions under which this result is derived are satisfied. The confidence limit for the Q-statistic is derived under the assumption that the sample vector $\mathbf{y}$ follows a multivariate Gaussian distribution. However, Jensen and Solomon point out that the Q-statistic changes little even when the underlying distribution of the original data differs substantially from Gaussian [12]. While we believe that normal traffic in our datasets is reasonably well described as multivariate Gaussian, we have not closely examined the data for violations of this assumption. However, we find that the Q-statistic gives excellent results in practice, perhaps due to the robustness noted by Jensen and Solomon.

An important property of this approach is that it does not depend on the mean amount of traffic in the network. Thus, one can apply the same test on networks of different sizes and utilization levels.

In Figure 4 we illustrate the effectiveness of subspace separation of $\mathbf{y}$ and $\tilde{\mathbf{y}}$ on two of our datasets. The upper half of the figures shows timeseries plots of $\|\mathbf{y}\|^2$ over week-long periods. On these plots, we have marked with circles the locations where volume anomalies are known to occur (based on inspection of the underlying flows, as will be described in Section 6.2). It is clear that the magnitude of the state vector $\mathbf{y}$ is dominated by effects other than the anomalies, and that it is quite difficult to see the effects of anomalies on the traffic volume as a whole.

In the lower half of each plot we show timeseries plots of the SPE, $\|\tilde{\mathbf{y}}\|^2$, over the same one-week periods. For each dataset, the values of the Q statistic $\delta_\alpha^2$ at the $1-\alpha$ = 99.5% and 99.9% confidence levels are also shown as dotted lines. The lower plots show how the projection of the state vector onto the residual subspace $\tilde{\mathcal{S}}$ very effectively captures the anomalous traffic while capturing little normal traffic, and so makes the statistical detection of anomalies much easier.

This figure shows how sharply the subspace method is able to separate anomalous traffic patterns (lower plots) from the mass of traffic (upper plots). It also gives some insight into why (as we will show in Section 6) the method yields such high detection rates combined with low false alarm rates. As can be seen in the lower plots, the distinct separation of anomalies from normal traffic means that almost all anomalies result in values of $\|\tilde{\mathbf{y}}\|^2$ greater than $\delta_\alpha^2$, while very few of the normal traffic measurements yield $\|\tilde{\mathbf{y}}\|^2$ greater than $\delta_\alpha^2$.

## 5.2 Identification

In the subspace framework, a volume anomaly represents a displacement of the state vector $\mathbf{y}$ away from $\mathcal{S}$. The particular direction of the displacement gives information about the nature of the anomaly. Thus our general approach to anomaly identification is to ask which anomaly out of a set of potential anomalies is best able to describe the deviation of $\mathbf{y}$ from the normal subspace $\mathcal{S}$.

We denote the set of all possible anomalies as $\{\mathcal{F}_i, i = 1, ..., I\}$. This set should be chosen to be as complete as possible, because it defines the set of anomalies that can be identified.

For simplicity of exposition, we will consider only one-dimensional anomalies; that is, anomalies in which the additional per-link traffic can be described as a linear function of a single variable. However, in Section 7 we show that it is straightforward to generalize the approach to multi-dimensional anomalies.

Then each anomaly $\mathcal{F}_i$ has an associated vector $\theta_i$ which defines the manner in which this anomaly adds traffic to each link in the network. We assume that $\theta_i$ has unit norm, so that in the presence of anomaly $\mathcal{F}_i$, the state vector $\mathbf{y}$ is represented by

$$\mathbf{y} = \mathbf{y}^* + \theta_i f_i$$

where $\mathbf{y}^*$ represents the sample vector for normal traffic conditions (and which is unknown when the anomaly occurs), and $f_i$ represents the magnitude of the anomaly.

Given some hypothesized anomaly $\mathcal{F}_i$, we can form an estimate of $\mathbf{y}^*$ by eliminating the effect of the anomaly, which corresponds to subtracting some traffic contribution from the links associated with anomaly $\mathcal{F}_i$. The best estimate of $\mathbf{y}^*$ assuming anomaly $\mathcal{F}_i$ is found by minimizing the distance to the normal subspace $\mathcal{S}$ in the direction of the anomaly:

$$\hat{f}_i = \arg\min_{f_i} \|\tilde{\mathbf{y}} - \tilde{\theta}_i f_i\|$$

where $\tilde{\mathbf{y}} = \tilde{\mathbf{C}}\mathbf{y}$ and $\tilde{\theta}_i = \tilde{\mathbf{C}}\theta_i$. This gives $\hat{f}_i = (\tilde{\theta}_i^T\tilde{\theta}_i)^{-1}\tilde{\theta}_i^T\tilde{\mathbf{y}}$. Thus the best estimate of $\mathbf{y}^*$ assuming anomaly $\mathcal{F}_i$ is:

$$\begin{aligned}\mathbf{y}_i^* &= \mathbf{y} - \theta_i\hat{f}_i \\ &= \mathbf{y} - \theta_i(\tilde{\theta}_i^T\tilde{\theta}_i)^{-1}\tilde{\theta}_i^T\tilde{\mathbf{y}} \\ &= (\mathbf{I} - \theta_i(\tilde{\theta}_i^T\tilde{\theta}_i)^{-1}\tilde{\theta}_i^T\tilde{\mathbf{C}})\,\mathbf{y} \end{aligned} \quad (1)$$

To identify the best hypothesis from our set of potential anomalies, we choose the hypothesis that explains the largest amount of residual traffic. That is, we choose the $\mathcal{F}_i$ that minimizes the projection of $\mathbf{y}_i^*$ onto $\tilde{\mathcal{S}}$.

Thus, in summary, our identification algorithm consists of:

1. for each hypothesized anomaly $\mathcal{F}_i$, $i = 1, ..., I$, compute $\mathbf{y}_i^*$ using Equation (1)
2. choose anomaly $\mathcal{F}_j$ as $j = \arg\min_i \|\tilde{\mathbf{C}}\mathbf{y}_i^*\|$.

As discussed in Section 2.2, in this paper we consider only the set of anomalies that arise due to unusual traffic in a single OD flow. Thus the possible anomalies are $\{\mathcal{F}_i, i = 1, ..., n\}$ where $n$ is the number of OD flows in the network. In this case, each anomaly adds (or subtracts) an equal amount of traffic to each link it affects. Then $\theta_i$ is defined as column $i$ of the routing matrix $\mathbf{A}$, normalized to unit norm: $\theta_i = \mathbf{A}_i/\|\mathbf{A}_i\|$.

## 5.3 Quantification

Having formed an estimate of the particular volume anomaly, $\mathcal{F}_i$, we now proceed to estimate the number of bytes the constitute this anomaly.

The estimated amount of anomalous traffic on each link due to the chosen anomaly $\mathcal{F}_i$ is given by

$$\mathbf{y}' = \mathbf{y} - \mathbf{y}_i^*.$$

Then the estimated sum of the additional traffic is proportional to $\theta_i^T\mathbf{y}'$. Since the additional traffic flows over multiple links, one must normalize by the number of links affected by the anomaly.

In the current case, where anomalies are defined by the set of OD flows, our quantification relies on $\mathbf{A}$. We use $\bar{\mathbf{A}}$ to denote the routing matrix normalized so that each column of $\mathbf{A}$ has unit sum, that is: $\bar{\mathbf{A}}_i = \frac{\mathbf{A}_i}{\sum \mathbf{A}_i}$. Then given identification of anomaly $\mathcal{F}_i$, our quantification estimate is:

$$\bar{\mathbf{A}}_i^T\mathbf{y}'.$$

## 5.4 Necessary and Sufficient Conditions for Detectability

Some anomalies may lie completely within the normal subspace $\mathcal{S}$ and so cannot be detected by the subspace method. Formally, this can occur if $\tilde{\mathbf{C}}\theta_i = \mathbf{0}$ for some anomaly $\mathcal{F}_i$. In fact this is very unlikely as it requires the anomaly and the normal subspace $\mathcal{S}$ to be perfectly aligned. However, the relative relationship between the anomaly $\theta_i$ and the normal subspace can make anomalies of a given size in one direction harder to detect than in other directions.

A sufficient condition for detectability in our context is given in [5]. Specializing their results to the case of one-dimensional anomalies, we can guarantee detectability of anomaly $\mathcal{F}_i$ if:

$$f_i > \frac{2\delta_\alpha}{\|\tilde{\mathbf{C}}\theta_i\|}$$

If a single-flow anomaly $\mathcal{F}_i$ consists of $b_i$ additional or missing bytes, then this threshold becomes

$$b_i > \frac{2\delta_\alpha}{\|\tilde{\mathbf{C}}\theta_i\|\,\|A_i\|}$$

This shows that, the larger the projection of the normalized anomaly vector in the residual subspace, the lower the threshold for detectability at a given confidence level.

In practice, the normal subspace $\mathcal{S}$ will tend to capture the directions of maximum variability in the data, meaning that it will tend to be more closely aligned with those flows that have the largest variances. Thus, anomalies of a given size will tend to be harder to detect in flows with large variance, as compared to flows with small variance. We quantitatively explore this effect for our data in Section 6.3.

# 6. VALIDATION

In this section, we evaluate the subspace anomaly diagnosis method using the datasets introduced in Section 3.

## 6.1 Methodology

Our validation approach is centered on answering two questions: (1) how well can the method diagnose actual anomalies observed in real data? and (2) how does the time and location of the anomaly affect performance of the method?

To answer the first question, we proceed as follows: using timeseries analysis on OD flow data, we first isolate a set of "true" anomalies. This allows us to then evaluate the subspace method quantitatively. In particular, it allows us to measure both the detection probability and the false alarm probability.

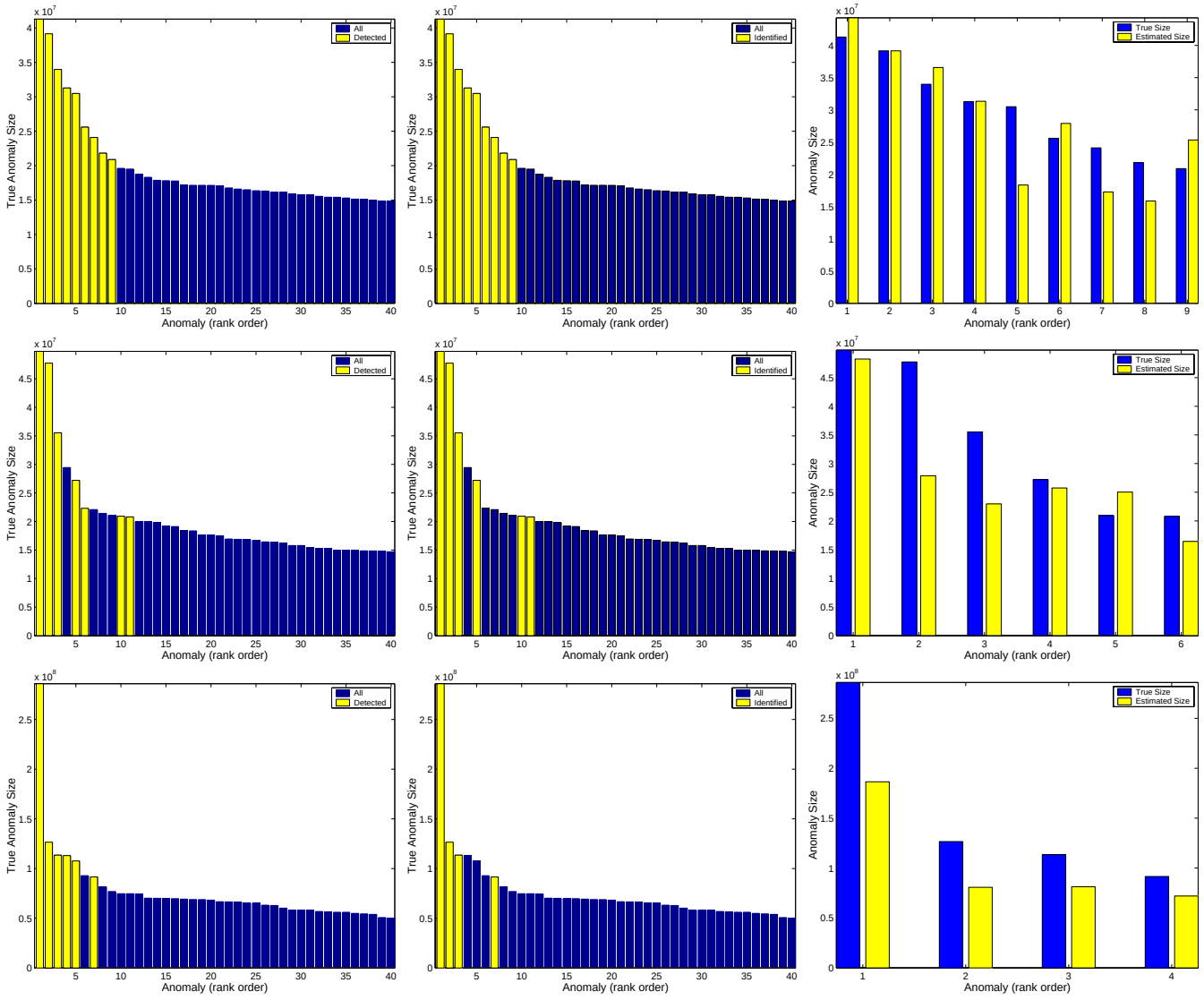
To answer the second question, we injected anomalies of different sizes in OD flows and applied our procedure to diagnose these known anomalies from link data. We perform this repeatedly for each timestep and for each anomaly so as to form a complete picture of how diagnosis effectiveness varies with the time and location of the anomaly.

In each case, we quantify the performance of each step in our diagnosis procedure as follows. Detection success is measured by two metrics: the detection rate and the false alarm rate. The detection rate is the fraction of true anomalies detected. The false alarm rate is the fraction of normal measurements that trigger an erroneous detection. Identification success is captured in the identification rate, which is the fraction of detected anomalies that are correctly identified. Finally, quantification success is measured by computing the mean absolute relative error between our estimate and the true size of all the volume anomalies identified.

## 6.2 Actual Volume Anomalies

To identify the set of "true" anomalies in our data (as a precursor to our validation step), we look for unusual deviations from the mean in each OD flow. There are two general classes of techniques to detect such changes in a non-stationary timeseries. The first class of methods identifies anomalies in an online manner, based on gross deviations from forecasted behavior. Simple instances of such a strategy are the exponential weighted moving average (EWMA) and Holt-Winters forecasting algorithms, both used in [2, 14]; more sophisticated examples are ARIMA-based Box-Jenkins forecasting models [14]. A second class of methods process data in batches and are based on signal analysis techniques, such as the wavelet analysis scheme used in [1]. Such schemes model the timeseries mean by isolating low-frequency components; anomalies are then flagged at those points in time that deviate significantly from the modeled behavior of the mean.

It should be pointed out that no scheme is ideal for isolating and quantifying spikes in a timeseries, and that some schemes are sensitive to a suite of configuration parameters and/or modeling assumptions. Thus, to obtain sets of true anomalies, we employed a candidate method from each class of techniques.




(a) Detection (b) Identification (c) Quantification

**Figure 5: Results of applying our method on link traffic to diagnose volume anomalies found in OD flows by the Fourier approach. From top to bottom: Sprint-1, Sprint-2, and Abilene.**

From the class of forecasting based techniques, we select EWMA and apply it on each individual OD flow timeseries to isolate anomalies. EWMA (also known as exponential smoothing) is a simple algorithm that predicts the next value in a given timeseries based on recent history. Specifically, if $z_t$ denotes the traffic in OD flow at time $t$, then the EWMA prediction for time $t+1$ is given by $\hat{z}_{t+1}$ as:

$$\hat{z}_{t+1} = \alpha z_t + (1-\alpha)\hat{z}_t$$

where $0 \leq \alpha \leq 1$ is a parameter that controls the relative weight placed on past values. We selected values for $\alpha$ based on applying a multi-grid parameter search (as in [14]) performed on sample training data, and found that values of $0.2 \leq \alpha \leq 0.3$ isolated the spikes in our data well. Anomalies can then be quantified by taking the difference between the forecasted and the actual value, *i.e.*, $|z_t - \hat{z}_t|$.[2]

From the class of signal analysis techniques, we draw on Fourier analysis to capture the diurnal trends that most OD flows exhibit over long periods of time. Specifically, we approximate the timeseries of each OD flow as a weighted sum of eight Fourier basis functions. These basis functions correspond to traffic variations with period 7 days, 5 days, 3 days, 24 hours, 12 hours, 6 hours, 3 hours, and 1.5 hours. The size of an anomaly then is the difference between the actual value and the modeled value, $|z_t - \hat{z}_t|$ where $\hat{z}_t$ is the Fourier approximation of the OD flow at time index $t$.

We confirmed that every OD flow anomaly that we could visually isolate was also discovered by both these approaches. However, we did find several instances where both schemes mistakenly marked a

[2]A problem with using such an EWMA-based approach (and most moving average schemes) to identify single-timestep spikes, is that the scheme often mistakenly marks the time after a spike as an additional spike. To avoid this problem, we ran EWMA in both directions, estimated the size of the spike in each direction and reported the minimum of the two estimates.

| Validation | Dataset | Anomaly Size | Detection | False Alarm | Identification | Quantification |
|---|---|---|---|---|---|---|
| Fourier | Sprint-1 | $2.0 \times 10^7$ | 9/9 | 1/999 | 9/9 | 15.6% |
| Fourier | Sprint-2 | $2.0 \times 10^7$ | 7/11 | 0/997 | 6/7 | 21.0% |
| Fourier | Abilene | $8.0 \times 10^7$ | 5/6 | 10/1002 | 3/5 | 33.0% |
| EWMA | Sprint-1 | $2.0 \times 10^7$ | 4/5 | 6/1003 | 4/4 | 16.8% |
| EWMA | Sprint-2 | $2.0 \times 10^7$ | 3/4 | 4/1004 | 2/3 | 20.8% |
| EWMA | Abilene | $8.0 \times 10^7$ | 2/3 | 13/1005 | 1/2 | 17.7% |

**Table 2: Results from actual volume anomalies diagnosed, all at 99.9% confidence level.**

point in time as an anomaly (*e.g.*, because of irregular periodicities in the underlying OD flow.) We did *not* remove these erroneous anomalies, so as to avoid introducing any bias in the set of anomalies. As a result there are some misidentified anomalies in our set of "true" anomalies, and in the text that follows we note where these affect our results.

We now present the results of evaluating our method against the set of volume anomalies obtained by applying the EWMA and Fourier methods. Recall that while both Fourier and EWMA are applied on the OD flow level, our diagnosis method operates on the link data.

In Figure 5(a) (left hand column) we present the top 40 anomalies isolated by the Fourier scheme for our three datasets. The first point to note is that in each dataset, there is a sharp knee in the rank-ordered plot of anomaly sizes. This suggests that the largest anomalies are qualitatively different from the large set of nearly equal anomalies to the right of the knee. In fact, the question of how large a spike should be in order to be considered an anomaly can be addressed in this way. We choose the anomalies that "stand out" to the left of the knee as the important set to detect. For the Sprint datasets this means that anomalies of greater than $2 \times 10^7$ bytes are important to detect; for the Abilene dataset, the value is $8 \times 10^7$ bytes.

Figure 5(a) also shows the performance of the subspace detection method, applied with a 99.9% confidence limit. (Detected anomalies are the light bars.) The plots show that above the cutoff value, the subspace method detects and identifies nearly every anomaly.[3] In addition, virtually none of the spikes below the cutoff (the non-anomalies), were detected; thus the scheme has low false alarm rates.

Figure 5(b) shows the same set of anomalies, but the light bars here indicate those anomalies for which the underlying flow was successfully identified. It shows that nearly every anomaly that was detected was successfully identified. (We did not attempt identification on anomalies that were not detected.) The accuracy of these results are consistent across all three datasets.

Finally, Figure 5(c) compares the anomaly size to the estimated size obtained via quantification, for the successfully identified flows. These results are sensitive to error in anomaly size determination in the Fourier analysis, so actual performance may in fact be better than what is shown here. Nonetheless, the results show that quantification can in general estimate the anomaly size reasonably accurately.

In Table 2 we report more detailed diagnosis results from both validation methods. To interpret these numbers it is useful to note that each dataset has 1008 points in time. For Sprint-1, only 9 of these 1008 timepoints are anomalies, all of which are detected and identified correctly, with only one false alarm reported. In fact, visually inspecting the false alarms revealed that most of them were classified as such because Fourier and EWMA underestimated the size of the anomaly. A reason for the higher false alarm rates in Abilene is that the nature of traffic (sampled 5-tuple IP flows) collected from this network is generally more noisy and so anomalies are more often missed by the smoothing of Fourier and EWMA. Therefore the actual rates might be better in reality.

The table confirms quantitatively that identification and quantification are reasonably accurate across all our datasets. Average quantification estimates for all but one dataset are within 21% of the true size of the anomaly, which is sufficiently accurate for operational settings [25].

To summarize, our results here demonstrate that regardless of the dataset or the validation method employed, the subspace diagnosis procedure shows high detection rates and low false alarm rates.

## 6.3 Synthetic Volume Anomalies

A second approach to quantify the effectiveness of the subspace method is to evaluate it on anomalies injected into the OD flows of Sprint-1 and Abilene.

We first select two injection sizes, *large* and *small*. The large injections are set to match the largest anomalies seen in the each dataset (*i.e.*, values above the cutoff in Figure 5). The small injections are set at values slightly below the cutoff in Figure 5 and so constitute non-anomalies that we *do not* want to detect. This allows us to quantify false alarm rates across all timepoints and all OD flows. For Sprint-1, we set large at $3 \times 10^7$ bytes and small at $1.5 \times 10^7$ bytes. For Abilene, we set large at $1.2 \times 10^8$ bytes and small at $5 \times 10^7$ bytes.

Then, in multiple experiments, we insert a spike of each size in every OD flow and at every point in time over the period of a day. For each permutation of spike size, timestep and OD flow selected, we generate the corresponding set of link traffic counts. We then apply our procedure and note whether it successfully diagnoses the injected anomaly.

The first set of these results are presented in Figure 6 which shows mean detection rates (rates are computed over OD flows) as a timeseries for the large injections in Sprint-1. This plot shows that the method's detection rate is fairly constant, regardless of when the anomaly was injected. Thus the diagnosis procedure works equally well across time, and is not affected by the underlying non-stationarity in traffic.

Evaluation results across OD flows are presented in Figure 7, which shows the histogram of detection rates (rates are computed over time) for both sized injections for Sprint-1. Figure 7(a) shows that the method detects the large (and therefore the most important) injections very well. On the other hand, Figure 7(b) shows that the small injected spikes rarely trigger detections, which is as desired. Together, these plots confirm that the subspace method has a high detection and a low false alarm rate.

In Figure 8, we present a scatter plot of mean detection rate (rates

[3]The missing anomaly at bar 4 of Sprint-2 was detected at the 99.5% confidence level. This was a case where Fourier overestimated the size of the anomaly.

| Network | Injection Size | Detection | Identification | Quantification |
|---|---|---|---|---|
| Sprint | Large ($3.0 \times 10^7$) | 93% | 85% | 18% |
| Abilene | Large ($1.2 \times 10^8$) | 90% | 69% | 21% |
| Sprint | Small ($1.5 \times 10^7$) | 15% | 14% | 11% |
| Abilene | Small ($5.0 \times 10^7$) | 5% | 3% | 18% |

**Table 3: Results on diagnosing synthetic volume anomalies.**

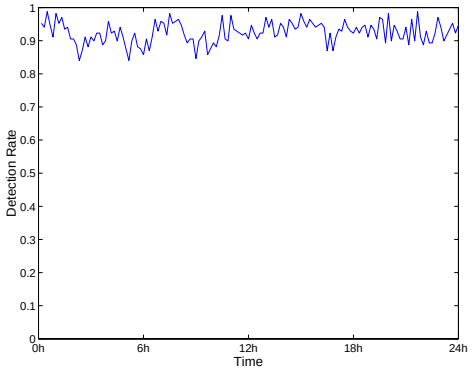
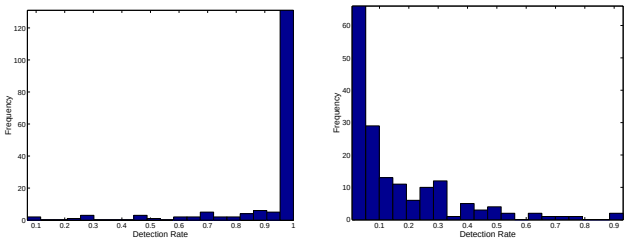



**Figure 6: Timeseries of detection rates on large injections (Sprint-1).**




(a) Large Injected Spike (b) Small Injected Spike

**Figure 7: Detection rate histograms from injecting synthetic spikes (Sprint-1).**

are computed over time) against the mean OD flow rate for the large injections for Sprint-1. This plot shows that for a fixed size anomaly, the method tends to detect the injections on the smaller OD flows better than on larger OD flows. There are two reasons for this. First, the larger variance OD flows are likely better aligned with $\mathcal{S}$, as explained in Section 5.4. Second, an inserted spike can be canceled out by a large negative spike in the OD flow, an effect more likely to occur on a high variance OD flow.

Corresponding experiments for Abilene and for identification and quantification yield similar results. Summary results from all these experiments for both Sprint-1 and Abilene are presented in Table 3. The first two rows quantify the method's ability to diagnose large injections, and shows very good detection, identification and quantification rates. The next two rows capture the method's ability to avoid the small injected spikes, representing false anomalies. These results show that regardless of the underlying network, OD flow and the time of the injected spike, the subspace method is able to diagnose the volume anomaly with high accuracy and low false alarm rates.

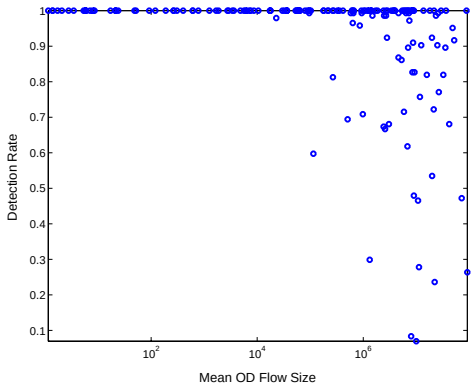



**Figure 8: Scatter plots of detection rate of large injections and mean OD flow rate (Sprint-1).**

# 7. DISCUSSION

In this section we consider issues related to deployment of the subspace method and its potential extensions.

## 7.1 Computational Complexity

Since the subspace method operates on link measurements, it imposes relatively low cost in data collection. We envision that the method may therefore be used as a first-level online monitoring tool, capable of raising an alarm and directing attention to particular OD flows as the focus of attention. The diagnosis of an anomaly may then trigger sophisticated, fine-scale (but more expensive) measurements to further isolate the cause of the anomaly.

To be used in this manner, the method must impose low computational demand. Computing all the principal components of link traffic $\mathbf{Y}$ is in fact equivalent to solving the symmetric eigenvalue problem for the covariance matrix, $\mathbf{Y}^T\mathbf{Y}$. The standard procedure for this relies on computing the singular value decomposition (SVD) of $\mathbf{Y}$. The computation complexity of a complete SVD of a $t \times m$ matrix is $O(tm^2)$ [9]. However, the computation is typically not expensive for reasonable-sized datasets. For our $1008 \times 49$ matrices the computation requires less than two seconds on a 1.0 GHz Intel-based laptop.

To apply the subspace method online, one processes each arrival of new traffic measurements using the matrix $\mathbf{P}\mathbf{P}^T$, which is derived from the SVD. Previous work has shown that (for OD flows) this matrix can be reasonably stable from week to week [16]. Thus one need only compute the SVD occasionally, rather than at each timestep.

Finally, it is conceivable that the straightforward SVD procedure could become a bottleneck if applied to data with a larger set of sources and destinations, *e.g.*, IP-level flow data. However, such cases may still be manageable using one of the methods that exist for updating previously computed decompositions as new data arrives [20].

## 7.2 Extensions

There are a number of ways in which the work described in this paper can be extended.

The choice made here to identify only single-flow anomalies is in fact somewhat arbitrary. An important generalization is to the case in which an anomaly involves multiple flows having different traffic intensities. This can occur when an anomaly involves multiple OD flows, when it arises from routing changes, or when it arises from network abuse (*e.g*, DDoS attacks). To handle this case, we may proceed as follows (see also [5]). We replace $\theta_i$ with a matrix $\Theta_i$ having as many columns as there are flows that participate in the anomaly; each column of $\Theta_i$ consists of the (normalized) column of $\mathbf{A}$ corresponding to a participating flow. We then replace $f_i$ with a vector $\mathbf{f}_i$ which will capture the intensity of the anomaly in each flow. The identification algorithms remain the same (in particular, Equation (1) is unchanged).

It is also possible to consider applying the subspace method to other metrics on links for which the $\ell_2$ norm is an appropriate measure. For example, we may choose to look at the number of IP flows passing over a link, or the average packet size. We believe that the method we have outlined in this paper is applicable to such metrics as well. In [15], we extend the subspace method to diagnose anomalies in a broader variety of traffic data.

## 7.3 Alternate Basis Sets for Y

Seen in a general light, the subspace method constructs an alternate basis set for representing link measurements that makes the separation of "normal" and "anomalous" conditions easier. It does so by making use of *spatial* correlation (correlation across links) in network measurements.

The same general strategy of seeking a useful alternate basis set for data representation lies behind previous approaches to anomaly detection as well [1, 2, 14]. However, these previous approaches have made use of correlation in the *temporal* domain – for example, using the wavelet transform, or using exponential smoothing.

However it is reasonable to ask how temporal correlation compares to spatial correlation as a tool for constructing such alternate basis sets in the multivariate case. This is an especially appropriate question since we made use of time-domain methods in constructing our "true" anomalies in Section 6.2.

For example, one method we used for constructing "true" anomalies was extracting the periodicity of OD flows (especially daily variation) from each OD flow using frequency-domain filtering via Fourier analysis. Since each link's traffic is a linear combination of traffic in all flows, we might consider that the same frequencies filtered from each link timeseries might yield good results for detecting anomalies in link data.

While a detailed exploration of this issue is beyond our scope, we contrast here the results of extracting "normal" link behavior via frequency-domain filtering, exponential smoothing, and subspace separation. To do this, we apply to each timeseries of link data the same Fourier-based filtering and EWMA smoothing that we applied to flow data in Section 6.2. In each case, this allows the separation of link measurements $\mathbf{y}$ into modeled and residual components. We then plot the squared norm of the residual vector as a function of time. These plots are shown in Figure 9. In this figure, the upper plot (showing the subspace method results) is the same as the lower plot in Figure 4(a).

The figure shows that the anomaly detection problem is much easier using the subspace method than using the EWMA and Fourier-based methods. In the case of the subspace method, it is possible to find a threshold such that all anomalies appear above the threshold, while very few normal data points appear above the threshold –

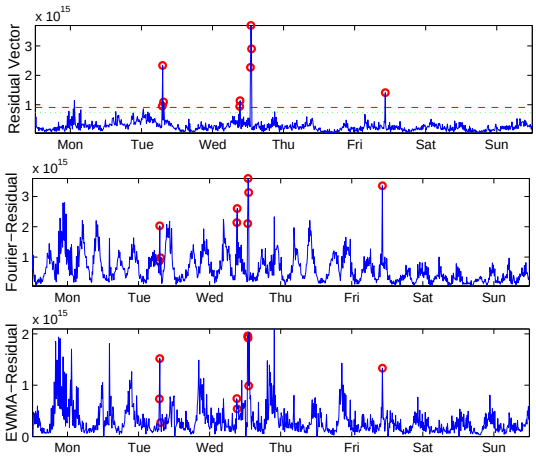



**Figure 9: Comparing squared magnitude of Subspace, Fourier and EWMA residual vectors.**

in other words, achieving high detection probability and low false alarm rate. The EWMA and Fourier methods are quite different; the anomalous data points do not stand sharply out from the normal data, and there is no threshold having high detection probability that does not also have high false alarm rate.

It is also striking that noticeable periodic behavior remains, even when the most significant frequencies are removed. This shows that periodic behavior in network traffic can be rather complex and difficult to model using only a small set of frequencies. The subspace method avoids this difficulty by not being restricted to a fixed set or number of frequencies. Quite complex behavior can be captured in the first few principal components, as shown in Figure 3(a).

Finally, it is important to note that temporal and spatial correlation are both valuable sources of predictability in network traffic, and there is room for methods that attempt to exploit both simultaneously. In particular, it is possible to use the subspace method across multiple time scales by applying PCA to the wavelet transform of measured data [19]. In principle, such a method can allow the detection of anomalies at all timescales.

# 8. RELATED WORK

A number of techniques have been proposed to detect anomalies in traffic volume. Some examples include [1, 2, 7, 14, 21]. As discussed in Section 7, all these schemes operate on *single-timeseries* traffic, measured for example from a network link, and independent of traffic on other links in a network. Thus, these techniques exploit temporal patterns within a single traffic timeseries to expose anomalies. In contrast, our scheme exploits correlation properties across links to detect *network-wide* anomalies.

Another distinguishing feature of our approach is that it does not require detailed modeling assumptions about the underlying normal traffic behavior. Previous methods have principally relied on timeseries models to approximate normal traffic [2, 10, 14] or have relied on building models of the data during a training period [24]. For example, the commonly used Holt-Winters algorithm builds a model for normal traffic based on parameter settings and *a priori* knowledge of the periodic structure in traffic. In contrast, our scheme does not rely on any parameter settings and normal traffic behavior is captured directly in the data, by extracting the common features across all links.

Isolating faults in networks (which is a more general problem than traffic anomaly detection) has also attracted some attention in

the past. A review of research in this area is [17]; noteworthy examples include [10] and [13]. The approach of [10] relies on an autoregressive (AR) model of signals, coupled with Bayes nets for detecting faults, but no significant validation is given. The methods of [13] rely on an exhaustive analysis of how faults might be detected through a variety of matching algorithms. Once a fault is found, the best explanation for it is obtained by relying on heuristics, some of which are unsubstantiated. In contrast, our approach is more systematic, relying on statistical tools; it requires no modeling assumptions, and has been successful validated on data from two modern backbone networks.

The problem of identification in our anomaly diagnosis framework bears resemblance to the problem of traffic matrix estimation as formulated in [23]. However, traffic matrix estimation is concerned with the problem of estimating *all* underlying OD flow intensities from link data, which is a much harder problem. As a result, most traffic matrix estimation work to date has concentrated on estimating mean values of large flows [18, 25]. In contrast, our work addresses estimation of anomalous values occurring in any flow.

The authors in [4] study a problem similar to the volume anomaly diagnosis problem and propose a solution method based on Bayesian models of anomaly events, but no evaluation is given.

Finally, as has been noted earlier in the paper, our methodology draws strongly on techniques from subspace-based fault diagnosis, such as those used in chemical engineering [5, 6]. Those methods exploit correlation patterns among state variables in an industrial process control setting, whereas we focus on covariance patterns across link traffic timeseries.

# 9. CONCLUSIONS

In this paper we have proposed an approach called the subspace method to diagnose network-wide traffic anomalies. The method can detect, identify and quantify traffic anomalies. The subspace method uses Principal Component Analysis to separate network traffic into a normal component that is dominated by predictable traffic, and an anomalous component which is more noisy and contains the significant traffic spikes.

We evaluate the method on volume anomalies, which are a specific instance of network-wide traffic anomalies resulting from unusual changes in the traffic of an OD flow. We showed how to use our method to diagnose volume anomalies from simple and readily available link measurements. We quantified the efficacy of our approach on OD flow data collected from two backbone networks, and showed that the subspace method can successfully diagnose volume anomalies with high detection rate and small false alarm rate.

Although the evaluation in this paper is in terms of volume anomalies, the subspace method is not specific to them. The strength of the method lies in its use of correlations in timeseries data from multiple links. This allows it to treat diagnosis of network-wide traffic anomalies as a *spatial* problem. Such an approach can therefore be extended to other types of network-wide data, and thus other types of network-wide anomalies.

Our ongoing work is centered on extending the methodology proposed here to diagnose additional network-wide anomalies, including routing related anomalies. Once effective mechanisms for diagnosing anomalies are built, we can begin to understand the causes of the anomalies. Finally, we plan to incorporate these algorithms in a toolset that can be used by network operators to better diagnose, understand and ultimately prevent network-wide anomalies.

# 10. ACKNOWLEDGEMENTS

We are grateful to Rick Summerhill, Mark Fullmer (Internet 2), Matthew Davy (Indiana University) for helping us collect and understand the flow measurements from Abilene, and to Bjorn Carlsson, Jeff Loughridge (SprintLink) and Richard Gass (Sprint ATL) for instrumenting and collecting the Sprint NetFlow measurements. We also thank Tim Griffin and Gianluca Iannaccone for feedback on early drafts. Finally, we are grateful to Eric Kolaczyk for many helpful discussions, and for clarifying the issues behind the discussion in Section 7.3.

# 11. REFERENCES

[1] P. Barford, J. Kline, D. Plonka, and A. Ron. A Signal Analysis of Network Traffic Anomalies. In *Internet Measurement Workshop*, 2002.

[2] J. Brutlag. Aberrant Behavior Detection in Timeseries for Network Monitoring. In *USENIX Fourteenth Systems Administration Conference (LISA)*, 2000.

[3] NetFlow. At `www.cisco.com/warp/public/732/Tech/netflow/`.

[4] D. Donoho and G. Flesia and N. El-Karoui and M. Elad. Inferring Sparse Causes for Extreme Network Events. In *Network Modeling and Simulation Workshop at Dartmouth College*, 2002.

[5] R. Dunia and S. J. Qin. Multi-dimensional Fault Diagnosis Using a Subspace Approach. In *American Control Conference*, 1997.

[6] R. Dunia and S. J. Qin. A Subspace Approach to Multidimensional Fault Identification and Reconstruction. *American Institute of Chemical Engineers (AIChE) Journal*, pages 1813–1831, 1998.

[7] F. Feather, D. Siewiorek, and R. Maxion. Fault Detection in an Ethernet Network Using Anomaly Signature Matching. In *ACM SIGCOMM*, 1993.

[8] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True. Deriving Traffic Demands for Operational IP networks: Methodology and Experience. In *IEEE/ACM Transactions on Neworking*, 2001.

[9] G. Golub and C. F. V. Loan. *Matrix Computations*. The Johns Hopkins University Press.

[10] C. Hood and C. Ji. Proactive Network Fault Detection. In *IEEE INFOCOM*, 1997.

[11] J. E. Jackson and G. S. Mudholkar. Control Procedures for Residuals Associated with Principal Component Analysis. *Technometrics*, pages 341–349, 1979.

[12] D. R. Jensen and H. Solomon. A Gaussian Approximation to the Distribution of a Definite Quadratic Form. *J. of the American Statistical Association*, pages 898–902, 1972.

[13] I. Katzela and M. Schwartz. Fault Identification Schemes in Communication Networks. In *IEEE/ACM Transactions on Neworking*, December 1995.

[14] B. Krishnamurthy, S. Sen, Y. Zhang, and Y. Chen. Sketch-based Change Detection: Methods, Evaluation, and Applications. In *Internet Measurement Conference*, 2003.

[15] A. Lakhina, M. Crovella, and C. Diot. Characterization of Network-Wide Anomalies in Traffic Flows. Technical Report BUCS-2004-020, Boston University, 2004.

[16] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. D. Kolaczyk, and N. Taft. Structural Analysis of Network Traffic Flows. In *ACM SIGMETRICS*, 2004.

[17] A. A. Lazar, W. Wang, and R. Deng. Models and Algorithms for Network Fault Detection and Identification: A Review. In *ICC*, 1992.

[18] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic Matrix Estimation: Existing Techniques and New Directions. In *ACM SIGCOMM*, 2002.

[19] M. Misra, H. H. Yue, S. J. Qin, and C. Ling. Multivariate Process Monitoring and Fault Diagnosis by Multi-scale PCA. *Computers and Chemical Engineering*, pages 1281–1293, 2002.

[20] M. Moonen, P. Dooren, and J. Vandewalle. A Singular Value Decomposition Updating Algorithm for Subspace Tracking. *SIAM J. Matrix Anal. Appl.*, pages 1015–1038, 1992.

[21] M. Roughan, T. Griffin, M. Mao, A. Greenberg, and B. Freeman. Combining Routing and Traffic Data for Detection of IP Forwarding Anomalies (Poster). In *ACM SIGMETRICS*, 2004.

[22] Juniper Traffic Sampling. At `www.juniper.net/techpubs/software/junos/junos60/swconfig60-policy/html/sampling-overview.html`.

[23] Y. Vardi. Network Tomography: Estimating Source-Destination Traffic Intensities from Link Data. *J. of the American Statistical Association*, pages 365–377, 1996.

[24] N. Ye. A Markov Chain Model of Temporal Behavior for Anomaly Detection. In *Workshop on Information Assurance and Security*, 2000.

[25] Y. Zhang, M. Roughan, C. Lund, and D. Donoho. An Information-Theoretic Approach to Traffic Matrix Estimation. In *ACM SIGCOMM*, 2003.