SLAC-PUB-13399

Using End-to-End Bandwidth Estimates for Anomaly Detection beyond Enterprise Boundaries

Fida Hussain*, Umar Kalim*†, Noman Latif* and Syed Ali Khayam*
*School of Electrical Engineering & Computer Science (SEECS)
National University of Sciences and Technology (NUST), Islamabad Pakistan
{syed.fida, noman.latif, ali.khayam}@seecs.edu.pk
†SLAC National Accelerator Laboratory, Stanford University, CA, USA
kalim@slac.stanford.edu

***Abstract*—Many algorithms have been proposed in the last decade to detect traffic anomalies in enterprise networks. However, most of these algorithms cannot detect anomalies that occur beyond enterprise boundaries. Performance monitoring and anomaly detection on end-to-end Internet paths, although important for network operations, is challenging due to lack of access and control over intermediate network devices. In this paper, we propose an algorithm that detects anomalies or significant events on an end-to-end Internet path by monitoring the path's available bandwidth. We first evaluate existing algorithms on a comprehensive dataset of more than a million bandwidth measurements spanning three years. We show that incorporating the *typical* behaviour of a path in the process of anomaly detection improves accuracy. We therefore propose to filter noisy bandwidth measurements to extract the typical behaviour or baseline statistical distribution of a path's bandwidth. This baseline model is in turn leveraged in a generic decision-theoretic framework to provide timely detection of significant path events. We show that the proposed detector provides high accuracy and surpasses the accuracy of existing techniques.**

I. Introduction

WHILE designing the LHC Computing Grid (LCG) [1], the SLAC National Accelerator Laboratory (SLAC) undertook a project to measure and evaluate the end-to-end performance of Internet paths. This evaluation was designed to facilitate operational concerns of critical grid/network applications, such as those exchanging massive amounts of high-energy physics experiments' data over the Internet. SLAC's intention was to adapt existing or develop new algorithms to detect anomalies on high-speed end-to-end paths for LCG. Similarly, in the last few years, large enterprises as well as academic and research networks have scaled dramatically in terms of capacities, sizes, supported applications and services, and thus automated end-to-end anomaly[1] detection has become one of the primary concerns of network operators. While anomaly detection in aggregate enterprise-level network traffic has received significant research attention in the last decade [2]–[6], detection of anomalous events occurring beyond enterprise boundaries is largely unexplored.

We realize that end-to-end anomaly detection also: 1) facilitates network operations [7] as it helps identify and quantify network path changes and provides alerts and diagnosis about whether the faults lie with the path or the applications; 2) allows network and capacity planning [8] by providing achievable performance and by maintaining historical information on network and economic growth[2]; 3) provides better insight into the impact of network performance on applications and protocols[3]; and 4) can facilitate higher network throughput by acting as a feedback mechanism for congestion control algorithms.

In this paper, we concern ourselves with timely and accurate detection of anomalous events beyond enterprise boundaries. These events may be caused by equipment failures (end-host or router failure, link outage,) unusual–possibly malicious usage (flash crowds, high volume flows, etc.,) and uncharacteristic behavior (misconfigurations, fluttering in traffic routes, etc.) We observe that most existing network anomaly detectors, aimed at detecting anomalous traffic within enterprise networks, cannot be used as such for *end-to-end* Internet paths. This is because the parameters used by these detectors (e.g., traffic volume, port frequencies, connection success rate, etc.) are typically unavailable beyond enterprise boundaries. We argue that the challenge presented by these circumstances can be addressed by employing end-to-end bandwidth estimates as a representative measure of the change of the state of an Internet path. We then present a comparative analysis of the accuracy and detection delay of four existing algorithms which can be adapted to detect end-to-end anomalies by monitoring bandwidth fluctuations, namely: 1) the Plateau algorithm [13]; 2) the Adaptive Fault Detector [14]; 3) the Kalman filter based detector [15]; and 4) the Holt-Winters detector [16], [17].

In this context, SLAC undertook the Internet End-to-End

This work is supported in part by the Higher Education Commission (HEC) of Pakistan under the project 44-DDWP-2007 and the Mathematical and Computational Sciences Division under the U.S. Department of Energy at SLAC. SLAC is operated by Stanford University for the U.S. Department of Energy under contract DE-AC02-76SF00515.

The labeled data set used in this paper is available online at `http://confluence.slac.stanford.edu/display/IEPM/Decision+Theoretic+Approach`. Please consider this paper for the best paper award.

[1]We use the words *anomaly* and *event* interchangeably in this paper.

[2]For instance, the annual ICFA-SCIC reports [http://www.slac.stanford.edu/xorg/icfa/scic-netmon/] use end-to-end anomaly detection as a metric to measure the digital divide; these measurements show a strong correlation with a variety of economic and development indices defined by the UN.

[3]Services that can benefit from end-to-end anomaly detection vary from sophisticated software providing remote access to scientific instrumentation [9], [10] to adaptive protocols [11] and applications [12].

Work supported in part by US Department of Energy contract DE-AC02-76SF00515.

SLAC National Accelerator Laboratory, Menlo Park, CA 94025

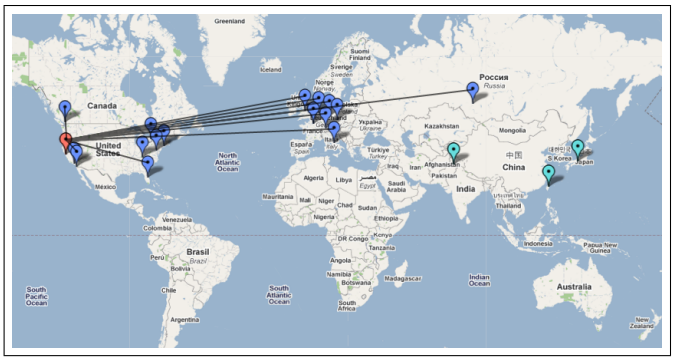


Fig. 1. World map of the IEPM monitoring sites. Typically each site performs performance evaluation tests to all the other sites. Here we label the site at SLAC which monitors all the other sites.

Performance Monitoring Bandwidth (IEPM-BW) [7] project in which end-to-end bandwidth estimates were collected. This dataset has more than a million bandwidth measurements collected over seventeen geographically diverse paths using different tools (iperf [18], pathChirp [19], and thrulay [20]) for a period of up to three years [7]. To establish the ground-truth for the IEPM-BW dataset, we develop an unbiased information-theoretic labeling algorithm using conditional entropy [21] based Markov chain analysis and verify the labeled anomalies against the available case studies.[4] We use this dataset for accuracy and delay evaluation of contemporary algorithms. We show that the performances of existing anomaly detectors (in terms of accuracy and detection delay) have a significant room for improvement. From the performance results of existing detectors, we note that an accurate path anomaly detector should incorporate and leverage the inherent statistical characteristics of *typical* bandwidth measurements observed on Internet paths.

To extract typical path characteristics from aggregate real-time data, we remove the noisy measurements by applying a low-pass median filter to the estimates. We then analyze the best-fit baseline distribution of the observed path. Since, in our study, we observed that a significant majority of the paths with bandwidth estimates in this dataset have either a Gaussian or a Weibull bandwidth distribution we leverage this baseline model of typical bandwidth measurements in a decision-theoretic likelihood ratio testing framework to detect anomalous events on an end-to-end Internet path.

The proposed decision-theoretic framework is generic and can be applied to data sets exhibiting any frequency distribution. Since the paths used in this study are Gaussian and Weibull, we provide mathematical formulations for these two distributions however models of other distributions may also be developed and applied in a similar manner. We use the Receiver operating characteristic (ROC) curves and detection delay to evaluate the accuracy and timeliness of the proposed detector. We show that the proposed detector provides high accuracy with low detection delay and surpasses the performance of existing techniques.

[4]Case studies are available at https://confluence.slac.stanford.edu/display/IEPM/Anomaly+Case+Studies.

TABLE I
PERFORMANCE MEASUREMENT TOOLS.

Tool	**Metric(s)**
Ping	Delay and loss
OWAMP [22]	One-way delay and loss
IPerf [18]	Achievable throughput
Thrulay [20]	Achievable throughput
Traceroute	Path
pathChirp [19]	Available bandwidth
Pathload [23]	Available bandwidth

TABLE II
BANDWIDTH MEASUREMENTS FROM SLAC TO THE MONITORED SITES, MARCH 2005 TO MARCH 2008

	Total Measurements (pathChirp)	**Anomalous Events**	**Duration of Events (Avg no. of measurements)** μ
utoronto.ca	40,614	38	41
desy.de	32,247	31	38
fzk.de	65,536	17	168
cern.ch	48,647	8	23
sdsc.edu	21,176	6	13
switch.ch	19,668	5	69
nslabs.ufl.edu	41,206	4	1035
triumf.ca	26,425	3	20
ornl.gov	35,339	2	32
nsk.su	20,117	1	8
dl.ac.uk	27,806	1	30
cacr.caltech.edu	61,871	1	59
infn.it	30,372	0	0
cesnet.cz	23,618	0	0
bnl.gov	23,580	0	0
anl.gov	17,968	0	0
ultralight.caltech	3,739	0	0

II. DATASET

Development of end-to-end event detection requires a comprehensive dataset of different measurements over an Internet path. Such a dataset requires an extensive monitoring system and SLAC started monitoring selected Internet paths since 1995 [24] (as part of the IEPM-BW [7] and the PingER project [8].) The purpose of the IEPM-BW project is to develop an infrastructure based on standard open technologies to make active end-to-end application and network performance measurements and predictions. The PingER project on the other hand now involves measurement to over 700 sites in over 160 countries and among them 50 are active monitoring nodes [25]. Since 2002, IEPM-BW has been providing an open repository of low impact network performance measurements – including delay, loss and connectivity information – to most of the Internet-connected world.

For the purpose of this study, we selected seventeen Internet paths shown in Fig. 1. These test sites of the IEPM-BW project include geographically diverse Academic and Research (A&R) institutes situated in Canada, Czech Republic, United Kingdom, France, Germany, Italy, Japan, Netherlands, Pakistan, Russia, Switzerland, Taiwan and USA; details of the network topology can be obtained from [7]. The reasons behind selecting these are: a) A variety of measurement tools are deployed at monitoring sites on either end of the paths; b) These paths feature minimum downtime and hence missing

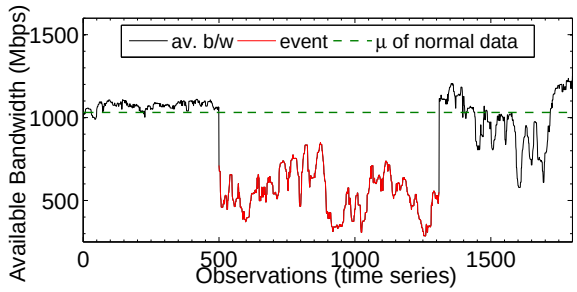


Fig. 2. Snapshot of approximately 2000 consecutive measurements spanning three hours and encompassing an anomaly. The observations tend to sustain themselves in the context. Note that the difference between the two subsets (anomalous and typical) lies in the mean and while the variance and the mean of one subset is approximately half of the other subset's mean.

data is not an impediment; c) They are spread over large geographical distances and represent important sections of the data grids; d) They feature diverse traffic content; and e) These paths form a feasible representation of typical Internet paths. Table I lists the performance metrics observed and the tools used by the IEPM-BW project. Some pertinent statistics about one of the tools, *pathChirp*, are shown in Table II[5]. Discussion of the last 3 columns of Table II is in the next section.

A. Available Bandwidth; Candidate Performance Metric

Any performance metric which exhibits sustained perturbations to reflect an anomaly can be considered as a good candidate for an anomaly detection algorithm. Parameters such as origin-destination (OD) flows, transport ports, frequency of connections between hosts, fields of the Ethernet, IP, TCP/UDP headers etc, although quite effective in detection of malicious traffic, are not available beyond enterprise boundaries. We thus turn to IEPM which provides us with end-to-end performance metrics i.e. round trip time (ms), one way delay (ms), packet loss (%), network path, available bandwidth (Mbps) and achievable throughput (Mbps). Network paths do not reflect state of the path. The possibility of ICMP traffic being classified as low priority traffic makes its estimates – round trip times and loss – unreliable. OWAMP, which measures one way delay, is not widely deployed at IEPM sites. However, available bandwidth is the metric which not only meets the assumptions for an anomaly detection algorithm, but also, IEPM has a wide deployment of reliable tools collecting measurements. Thus after extensive empirical studies[6] we selected average available bandwidth as the metric for our study because: a) available bandwidth estimation algorithms are mature and accurate [26]–[28]; b) bandwidth measurements remain stable during normal operations; c) they are perturbed throughout the course of an anomaly; and d) as observed by us and prior studies [26]–[28] achievable throughput measurements made by tools such as thrulay or iperf were highly inconsistent due to their inherent pipe filling nature. Alternatively, features of available bandwidth are outlined in Fig. 2 which shows that variations in available bandwidth are either significantly different from the typical behavior and/or persist for a noticeable duration. We also note that state of the art techniques [13], [14] also advocate the use of this metric.

The study [28], agrees with our understanding, and concludes that Pathchirp is an accurate tool for obtaining available bandwidth estimates without being intrusive. The alternative, pathload, influences the ongoing TCP sessions and makes them change their behaviour to accommodate the new flow – this is because of the inherent nature of TCP. This makes pathload inappropriate for our study.

III. Labelled Dataset

The process of evaluation of path anomaly detectors demands a labelled dataset with clearly demarcated anomalous boundaries/windows. Unfortunately such a labelling is not available for known end-to-end performance measurement datasets. Therefore here we present an unbiased, information theoretic labelling algorithm to annotate the IEPM-BW dataset.

1) Definition of a Path Anomaly: An accurate labelling algorithm caters for the baseline or typical behaviour of the bandwidth measurements which tends to sustain itself over a defined duration. The duration may be defined in terms of number of consecutive measurements. Thus, we define a *path anomaly* as:

Definition 1: A set of observations is called an *anomaly* if these (deviant) observations are *statistically significantly different* – in magnitude – from the typical observations of the Internet path and persist for a *minimum duration*.

To quantify this minimum duration we studied the temporal dependence of bandwidth measurements. Our intention was to identify the number of bandwidth measurements that the most recent observation depends upon. A well-known measure of dependence is conditional entropy [21]. Interestingly, as will be shown subsequently, evaluation of the conditional entropy of bandwidth measurements also yields a quantification of the magnitude of bandwidth perturbations that can be used to qualify a sequence of measurements as anomalous.

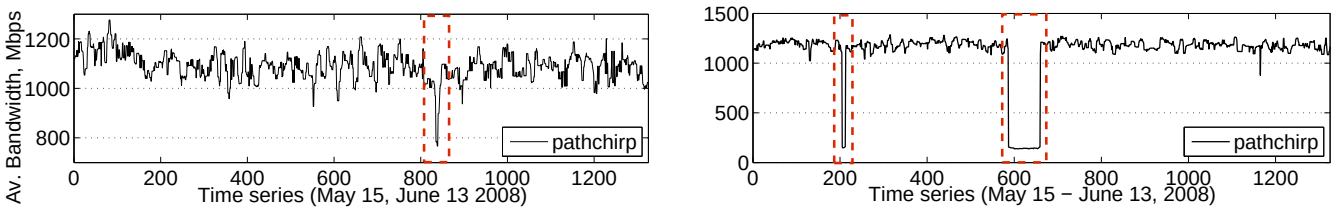
To identify the order of correlation present in the available bandwidth estimation random process, we define a Markov chain based stochastic model as follows. Let X_n be a time series of available bandwidth estimates. Let γ and κ be two parameters providing a confidence interval around the mean:

$$[\gamma \times E(X_n), \kappa \times E(X_n)], \tag{1}$$

where $E(X_n)$ is the expected value of X_n. Thus bandwidth values smaller than $\gamma E(X_n)$ or larger than $\kappa E(X_n)$ are classified as anomalous, while values falling in the above range are treated as typical values. Thus, to remove potentially anomalous bandwidth values, we increase γ and decrease κ (i.e., dilate the confidence interval) and apply a decimation filter by removing values from X_n that lie outside the confidence interval. This results in a decimated time series $X_n^{(\gamma,\kappa)}$ stochastic process. As mentioned earlier, $X_n^{(\gamma,\kappa)}$ with small γ

[5]Approximately the same number of thrulay and iperf bandwidth estimates were analyzed for each path; total number of analyzed (pathChirp+iperf+thrulay) measurements are approximately 1.1 million.

[6]Details are available on the project website.



(a) Deutsches Elektronen-Synchrotron, Germany

(b) San Diego Supercomputing Center, USA

Fig. 3. Sample of available bandwidth measurements as seen from SLAC with annotated anomalies.

and large κ will include more anomalous observations. For ease of notation, we denote $X_n^{(\gamma,\kappa)}$ as X_n in the following text.

The Markov chain model X_n is trained by first dividing all possible bandwidth values in multiple bins. Each bin then represents a state of the Markov chain, while the set of all bins $\psi^{(1)}$ is its state space. Based on this state representation we can define a first order Markov chain $X_n^{(1)}$ in which each bin represents a state of the random process. Conditional entropy of $X_n^{(1)}$ is:

$$H^{(1)} = -\sum_{i\in\psi^{(1)}}^{n} \pi_i^{(1)} \sum_{j\in\psi^{(1)}}^{n} \left(p^{(1)}_{X_{n=j}|X_{n-1}=i}\right) \log_2\left(p^{(1)}_{X_{n=j}|X_{i-1}=i}\right), \quad (2)$$

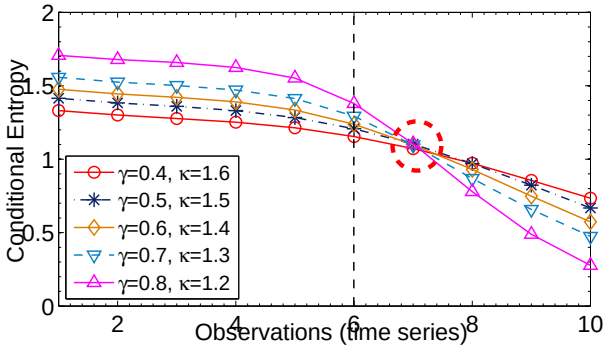
where $\pi_i^{(1)}$ is the average probability of being in state i. The measure $H^{(1)}$ defines to what extent average information is available in X_n when is it predicted using X_{n-1}. If the random variable is correlated with states before X_{n-1}, $E^{(1)}$ will be relatively large, implying that information about X_n not provided by X_{n-1} is high. In such a case, generalizing the above discussion, we can define a higher l^{th} order Markov chain, $X_n^{(1)}$, in which each state is an l-tuple $< i_0, i_1, ..., i_{l-1} >$ representing the values taken by the random process in the last l instances. Aggregating multiple instances in a single state allows us to satisfy the Markov property and hence a transition probability matrix $P^{(l)}$ can be computed by counting the number of times $< i_0, i_1, ..., i_{l-1} >$ is followed by $< i_0, i_1, ..., i_{l-1}, i_l >$. The conditional entropy of $X_n^{(l)}$ defined on ψ^l can then be computed using the same method as $H^{(1)}$.

It is easy to observe that $H^{(1)} < H^{(2)} < ... < H^{(l)}$ as for each random variable the previous can either be independent or provide some information about the current variable. Similarly the expected behaviour of conditional entropy for decimated time series $X_n^{(\gamma,\kappa)}$ would be:

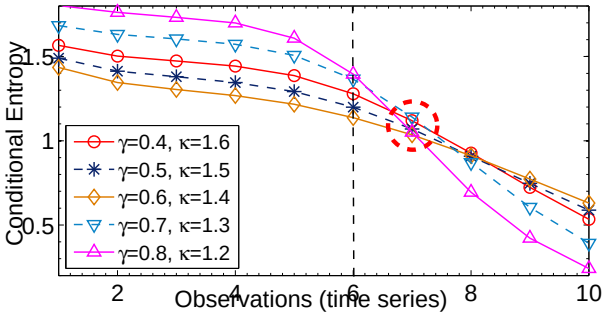
$$E^{(\gamma_k,\kappa_k)} \leq E^{(\gamma_{k-1},\kappa_{k-1})} \leq ... \leq E^{(\gamma_1,\kappa_1)}, \quad \gamma_k < \gamma_{k-1} \text{ and } \kappa_k > \kappa_{k-1}. \quad (3)$$

For small γ and large κ, the stochastic process will include both typical and anomalous observations. Hence, the order of correlation in X_n for such a decimated process can quantify the extent of temporal dependence in the bandwidth estimates during anomalous activity.

This can be observed in Fig. 4 for decimated measurements with large confidence intervals; i.e., small γ and large κ. It can be observed that for both the links analysed in this



(a) BNL as seen from SLAC



(b) DESY as seen from SLAC

Fig. 4. Conditional entropies of bandwidth time series decimated using different confidence intervals.

figure, values in high confidence intervals exhibit a consistent correlation trend up to 6 observations followed by a steep decaying trend. Thus, for a dataset with anomalous and typical measurements, consistent correlation is present up to 6 bandwidth measurements.

Interestingly, Fig. 4 also gives us an indication about the magnitude of a bandwidth fluctuation that should be classified as an event. Note the two distinct patterns around $\gamma \approx 0.5$ and $\kappa \approx 1.5$. For $\gamma \leq 0.5$ and $\kappa \geq 1.5$ dependence delay shows less steepness. Comparatively a steep decay structure is observed for $\gamma \geq 0.5$ and $\kappa \leq 1.5$. This can be explained as the degree of dependence is larger for a decimated time series $X_n^{(\gamma_m,\kappa_m)}$ having a combination of anomalous and typical values where $\gamma_k > \gamma_m$ and $\kappa_k < \kappa_m$. These two decay patterns show that bandwidth values outside the $[0.5 \times E\left(M\left(X_n\right)\right), 1.5 \times E\left(M\left(X_n\right)\right)]$ range can be classified as anomalous. So, now we can define the *significantly different* and *minimum duration* terms in Definition 1.

Definition 2: A subset of consecutive measurements A of

Algorithm 1: Labeling data with anomalies.

Data: a) Array of performance measurements Δ of length N, length τ of sliding window A and the duration τ for which an abnormal activity needs to persist before it is considered an event

Result: Ψ: Array of time brackets defining all independent events

for $\{\delta_i \in \Delta | 1 \leq i \leq N\}$ **do**
  Compute μ_A ;
  if $\left(0.5 \leq \frac{\mu_A}{\mu_\Delta} \leq 1.5\right)$ **then**
    Mark as typical observation;
  else Mark as an anomalous window and add to Ψ;
end
for $\{$*all alerts in* $\Psi\}$*;* **do**
  If required, coalesce alert-windows considering τ to identify unique observations with adjusted boundaries;
end

the dataset Δ is said to be significantly different if:

$$\frac{\mu_A}{\mu_\Delta} < 0.5 \text{ or } \frac{\mu_A}{\mu_\Delta} > 1.5, \tag{4}$$

where μ_A and μ_Δ represent the sample means of A and Δ, respectively.

Definition 3: The minimum number of values τ over which an anomaly should sustain itself are $\tau > 6$ values.

In an attempt to obtain empirical results and further consolidate our findings we plot the data sets as time series, Fig. 3. These plots substantiate the aforementioned results and led us to confirm the following observations: a) Anomalous events tend to exhibit sustained behaviour; b) These anomalous observations are typically 0.5μ or 1.5μ in magnitude; and c) A significant difference is observed in the mean of the typical and anomalous observations and not in their variance. This last observation is also illustrated in Fig. 2.

2) Labelling Algorithm: To label a dataset Δ, we first compute the mean μ_Δ of the dataset. We then analyse the measurements with a sliding window A of length τ values. The mean μ_A of the window is computed and a test is performed as per definition 2. Once all observations are scrutinized, windows marked as anomalous are analysed and coalesced[7] to identify the demarcations of unique anomalies. The detailed data labelling procedure is described in Algorithm 1 and number of events detected on each path are listed in Table II and last column presents the average number of measurements in the events. We have also analyzed the number of events detected in different datasets by varying the minimum number of values but beyond 6 values does not introduce a change in the number of identified events.

The events labelled using Algorithm 1 are verified against available anomalies case studies[8] to ensure correctness. Note that while the above data labelling algorithm is accurate, it cannot be used as an effective anomaly detector because it requires all bandwidth measurements to be available before the algorithm can start event classification. Consequently, this algorithm can only be used for offline data processing rather than real-time event detection.

3) Brief Discussion of Events: Table II clearly shows the diversity of paths used in this study. It can be seen that 4 out of the top 5 paths having the most number of events are situated outside the USA. Since these paths traverse international boundaries, changes in the capacity, configuration and/or policies of intermediate routers directly impact their bandwidth and delay characteristics. On the other hand, paths to government-owned sites inside the US generally have few, if any, anomalous events because these sites are generally available through rich and robust connectivity paths. It should also be highlighted that the average duration of anomalies varies considerably across different paths—ranging from 8 measurement long events to events that last more than a 1000 measurements. Due to this variation, timely detection becomes critical; otherwise, as will be elaborated in the next section, transient (short) anomalies can go undetected while late detection of persistent (long) anomalies will lead to undesirable detection delays. Furthermore, note that variations in anomaly durations on the same path are also quite high. The time- and space-dependent natures of end-to-end anomalies pose a significant challenge for a generic anomaly detection algorithm which is expected to detect events on *any* Internet path.

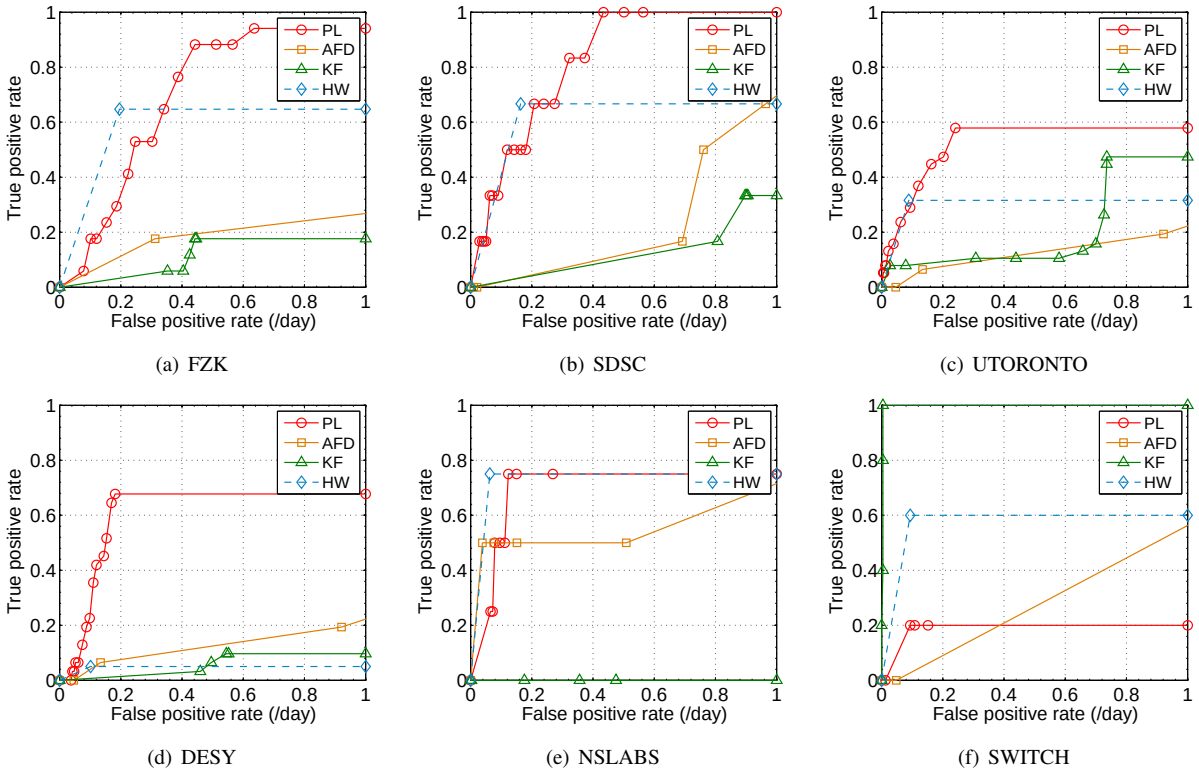
IV. Evaluation of Existing Internet Path Anomaly Detectors

Performance of an Internet path anomaly detector, which attempts to detect anomalies beyond enterprise boundaries, is defined by its accuracy (detection and false alarm rates) and the speed of event detection. More specifically, network traffic typically shows three types of variations [29]: 1) daily periodic behavior or diurnal patterns, 2) random and sporadic fluctuations, and 3) occasional bursts of high or low network activity. Since the first two types of variations do not warrant remedial measures, they are not *interesting* for network operators. The third type of traffic variation satisfies our definition of an event as it causes prolonged perturbations in an end-to-end path and therefore requires immediate attention [30]. The problem then is: When does an event being treated as uninteresting (diurnal or sporadic) become interesting? An inherent tradeoff between accuracy and delay can be observed here. If we wait long enough for more measurements to arrive before flagging the current measurements as anomalous, the accuracy in detecting interesting events will improve. However, such a procedure will lead to significant detection delays which are highly undesirable in the present problem. A good end-to-end anomaly detector should balance this accuracy-delay tradeoff.

In this section, we first evaluate the suitability of contemporary anomaly detection algorithms for end-to-end anomaly detection. We then use the IEPM dataset to evaluate the

[7]Note that each unique anomaly must be of a duration greater than τ. Also the separation between anomalies must be greater than τ to classify the anomaly as unique.

[8]Case studies are available at https://confluence.slac.stanford.edu/display/IEPM/Anomaly+Case+Studies.



(a) FZK (b) SDSC (c) UTORONTO

(d) DESY (e) NSLABS (f) SWITCH

Fig. 5. ROC curves of the Plateau Algorithm (PL), Adaptive Fault Detection (AFD), Kalman Filter method (KF) and Holt-Winters (HW) method for pathChirp measurements as seen from SLAC; for clarity, we truncate the x-axis at 1 false positive/day as the [0,1] range is reasonable for accuracy comparison. Note that due to lack of space we present six ROC curves here; ROC curves of all eight Internet paths are available at https://confluence.slac.stanford.edu/display/IEPM/Decision+Theoretic+Approach.

accuracy and timeliness of anomaly detection algorithms that can be adapted to use these end-to-end parameters.

A. Existing Anomaly Detectors

We investigated the suitability of several existing anomaly detectors [13]–[16], [31]–[37] for the present problem of anomaly detection beyond enterprise boundaries. From the study of these methods, we observed that most of these algorithms are designed to detect malicious anomalies in aggregate traffic and cannot be adapted to the present problem of event detection on end-to-end paths because: a) they rely on detection features and parameters that are not available beyond enterprise boundaries; b) they do not cater for non-malicious anomalies; and c) they cannot be modified to use parameters that are available on an end-to-end link–such as available bandwidth. For example, the maximum entropy and subspace detectors [31], [32] use the distributions of origin-destination (OD) flows and transport ports to mine anomalies. Rate-limiting and threshold random walk (TRW) algorithms [33]–[35] detect anomalies by measuring the frequency of connections between hosts. PHAD and NETAD [37] uses all 33 fields of the Ethernet, IP, TCP/UDP headers to calculate an anomaly score before raising an alarm. The parameters described above, although quite effective in detection of malicious traffic in aggregate enterprise traffic, are not available on end-to-end links due to lack of access and control over intermediate network devices.

B. End to End Anomaly Detectors

Most of the existing anomaly algorithms are designed to leverage a specific set of traffic features and, consequently, cannot be adapted to use the available bandwidth metric. We have identified the following anomaly detection algorithms which are generic enough to be adapted to the problem of end-to-end Internet path event detection: 1) the Plateau algorithm (PL) by Logg et al. [13]; 2) the Kalman filter (KF) based detector by Augustin et al. [15]; 3) the Adaptive Fault Detector (AFD) by Hajji [14]; and 4) the Holt Winters (HW) detector by Brutlag [16]. The rest of this section compares the accuracy and detection delays of these detectors; details of these detectors are skipped for brevity and interested readers are referred to the original papers for these details.

C. Accuracy and Delay Comparison

1) Accuracy Comparison using ROC Curves: We compare the accuracies of existing detectors using ROC curves [38]. The main performance evaluation metrics used in ROC curves of the present problem are defined below:

- A *true-positive* (TP) is the correct classification of an anomalous bandwidth event;
- A *false-positive* (FP) is the incorrect classification of a typical bandwidth measurement as anomalous;

- *TP rate* is the ratio of the correctly classified events to the total number of events present in a dataset;
- *FP rate* is the ratio of the incorrectly classified typical values to the total number of days observed.

ROC curves are drawn with the true-positive rate on the y-axis and the false-positive rate on the x-axis. Each point on the ROC curve represents performance results for one configuration (or threshold value) whereas the curve represents the behavior for the complete set of configurations. When compared, the steepest and highest curve is considered the best as it approaches the highest true-positive rate with the lowest false-positive rate.

To generate ROC curves, we varied the buffer lengths and the threshold values[9] of the algorithms. The ROC curves of the PL, AFD, KF, and HW algorithms are shown in Fig. 5. We observe that both the AFD and KF perform poorly with unacceptable FP rates. For instance, when applied to the Internet path SLAC-UTORONTO, the AFD method eventually achieves a TP rate of 1 (not shown in the figure,) but at the cost of 5.3 FPs per day. While the Holt-Winters method provides better accuracy than AFD and KF, its detection rate saturates at a particular point. The Plateau algorithm provides the best accuracy with high TP rates against relatively low FP rates. However, its FP rate at its highest detection point is quite high; ranging between 0.2 and 0.6 false positive per days (1 to 3 incorrect alarm in a five day period). Also, as shown in Fig. 5(c), in some cases the Plateau algorithm failed to achieve 100% detection rate within the 1 false positive/day constraint.

AFD has poor accuracy because it relies on the assumption that the difference between consecutive typical measurements is small. While this assumption holds for frequent bandwidth measurements, in case of measurements that are spread out in time (e.g., the IEPM-BW measurements every 30-45 mins,) large variations between consecutive measurements enhance the sensitivity and the FP rate of the AFD algorithm. Also, AFD assumes that the observed data has a k-variate Gaussian distributions which is not the case in present problem—as will be shown subsequently [Section V-B]. Kalman filter fails primarily because it assumes that bandwidth measurements are corrupted by an additive Gaussian noise process, an assumption that does not hold in the present context. The Holt-Winters based method performs poorly because it uses exponential smoothing while assuming that the input data exhibits explicit seasonal patterns which are not observed in the IEPM data set. Consequently, the HW detector considers noise as seasonality and therefore tries to fit seasonal patterns to the data set where none exists. Plateau provides better accuracy because, instead of making assumptions about the bandwidth or noise processes, it leverages the mean and standard deviation of the real-time bandwidth measurements for anomaly detection.

2) Delay Comparison: Detection delay is generally defined as the time taken by an anomaly detector in identifying an anomalous event. Since IEPM's measurements are made with regular intervals, we define detection delay as the difference between the first observation flagged as anomalous by an algorithm and the first actual anomalous observation of the event.

Fair comparison of detection delays is difficult because different detectors feature different FP and TP rates. Consider, for instance, an anomaly detector that classifies all bandwidth measurements as anomalous. Now while this detector is completely inaccurate, its detection delay will be zero. Therefore, fair comparison of detection delays requires that delay is computed for a practical point on the ROC curve. To this end, for each detector we select the ROC point of the detector having the maximum possible detection rate; PL in the present case. For the highest TP rate of the PL detector, the detection delays of all detected events are computed. For the remaining detectors, we compute detection delays at ROC points having similar FP rate as the PL detector. Average detection delays of all detectors are computed in terms of number of observations required before an event is detected. Also we define the detection delays of events not detected by an anomaly detector as ∞.

Delay results for the Internet paths between SLAC and UTORONTO, CERN, DESY, SDSC, TRIUMF, NSLABS, SWITCH and FZK are listed in Table III. PL, KF and the HW method provide similar detection delays, while the AFD method requires a significantly larger number of observations before an event is detected. We also observed with Plateau that reducing the size of the buffers results in a decrease in the detection delay but this has an adverse effect of making the algorithm sensitive to spurious changes which subsequently increase the algorithm's FP rate.

D. Discussion

Based on the results of this section, the accuracies of existing anomaly detectors leave significant room for improvement. Overall, we observed that all of the existing anomaly detectors are general-purpose anomaly detectors which are designed to flag changes in any underlying observation metric. We show in the following section that bandwidth measurements on Internet paths exhibit some very specific statistical characteristics that can facilitate classification. However, existing algorithms do not take these inherent bandwidth characteristics into account and are therefore unable to provide the required performance.

V. Bandwidth Statistics and the Decision-Theoretic Anomaly Detector

In this section, we first show that the baseline behavior of an Internet path's available bandwidth measurements exhibit a unique baseline distribution. Observations that deviate from this baseline distribution can thus be classified as anomalous. We then use the baseline model in a decision-theoretic framework for real-time anomaly detection.

A. Extracting Baseline Behavior of Bandwidth Measurements

Given a set of bandwidth measurements, extraction of the baseline behavior essentially entails removing all anomalous

[9]The range of values used to analyze the dataset and compile results presented in Section VI are posted at https://confluence.slac.stanford.edu/display/IEPM/Decision+Theoretic+Approach

TABLE III
AVERAGE DETECTION DELAY $\bar{t}$ (NUMBER OF OBSERVATIONS REQUIRED BEFORE AN EVENT IS DETECTED)

	Plateau Algorithm				**Adaptive Fault Detection**				**Kalman Filter Method**				**Holt Winters Method**				
	Detected		*Undetected*		*Detected*		*Undetected*		*Detected*		*Undetected*		*Detected*		*Undetected*		*Total*
	#	$\bar{t}$	#	$\bar{t}$	#	$\bar{t}$	#	$\bar{t}$	#	$\bar{t}$	#	$\bar{t}$	#	$\bar{t}$	#	$\bar{t}$	
UTOR	23	4.98	15	∞	4	53.25	34	∞	4	4.75	34	∞	12	10.33	26	∞	38
CERN	1	2.00	7	∞	0	-	8	∞	0	-	8	∞	3	9.33	5	∞	8
DESY	14	12.60	17	∞	1	47.43	30	∞	0	-	31	∞	0	-	31	∞	31
SDSC	1	0.0	5	∞	0	-	6	∞	0	-	6	∞	4	7.07	2	∞	6
FZK	7	14.27	10	∞	3	55.27	14	∞	0	-	17	∞	11	22.13	6	∞	17
TRIUMF	3	2.33	0	-	0	-	3	∞	0	-	3	∞	3	0.67	0	-	3
NSLABS	2	2.33	2	∞	2	45.67	2	∞	0	-	4	∞	3	1.57	1	∞	4
SWITCH	1	26.67	4	∞	3	7.63	2	∞	5	16.67	0	-	3	13.33	2	∞	5

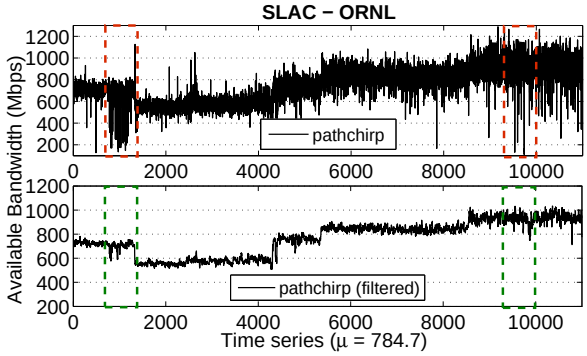


Fig. 6. Low-pass median filtering of bandwidth measurements to extract baseline behavior; the time series is annotated to show how median filtering results in removal of sustained anomalies and spurious measurements.

observations (and the corresponding bandwidth values) from the set. The remaining measurements can then be used to characterize the baseline behavior. Anomalous bandwidth values always cause significant fluctuations in the measurements, albeit these fluctuations may be sustained and spurious in nature. These two types of anomalies are shown in Fig. 3. Both of these anomalies should be removed from the dataset before baseline behavior is characterized.

To remove the anomalous bandwidth measurements from the dataset, we apply an n-tap median filter to the dataset. A median filter is a sliding window low-pass filter that stores n previous values of the input and at each step outputs the median of the stored values. Consequently, high frequency spikes are removed from the input data. Note that the value of n is a crude upper bound on the maximum duration of an anomaly. If a bandwidth change sustains itself beyond n observations then it is treated as a change in the baseline behavior. We define an empirical lower bound on n as:

$$n \geq 2\tau\upsilon,$$

where υ is the average number of IEPM performance measurements made in one hour and τ is the minimum duration over which an event should sustain itself which in this case is $\tau > 3$ hours. Through empirical evaluation, we observed that a value of $n = 15$ is sufficient to remove sustained and spurious bandwidth fluctuations from the present dataset without affecting the baseline characteristics. An example of the baseline bandwidth values extracted through median filtering is shown in Fig. 6.

B. Statistical Behavior of Available Bandwidth Measurements

We randomly selected sample subsets from the IEPM data to identify the baseline characteristics of the observed Internet paths. These sample subsets included observations made over three or more consecutive days. A window of three days was selected because we observed that anomalies in the IEPM dataset generally persisted for less than three days and conversely any change persisting beyond three days appeared to be permanent.

In more than two-thirds of the pathChirp subsets, we observed that the measurements follow either a Gaussian or a Weibull distribution. Examples of these subsets—with Gaussian and Weibull distributions—are shown in Table IV. This is an important statistical characteristic of the underlying typical (i.e., devoid of anomalies) bandwidth behavior which can and should be leveraged for baseline behavior characterization and subsequently for anomaly detection. We use this baseline behavior of available bandwidth measurements in a decision-theoretic anomaly detection framework in the next section.

Two important points should be emphasized here: 1) While the decision-theoretic framework presented henceforth is generic and can be applied to *any* bandwidth distribution, we only derive analytical expressions of and report results for Gaussian and Weibull distributions because they characterize the majority of the links being evaluated in this study; 2) Considering the data set and labeled anomalies, we only present results[10] for eight Internet paths which feature notable number of anomalies; results on the remaining paths are qualitatively similar and are therefore skipped for brevity.

C. Decision-Theoretic Model of Bandwidth Measurements

Let $\mathcal{R}_i$ denote the i-th available bandwidth measurement. These measurements are either the baseline behavior of the path (i.e., the internal response [39]) or comprise anomalous observations (i.e., the internal response modified by noise). We define two hypotheses: $\mathcal{H}_0$, the null hypothesis where $\mathcal{R}_i$ represents the internal response (i.e., the baseline characteristics); and $\mathcal{H}_1$, the alternate hypothesis where R_i represents the internal response modified by noise (i.e., anomalous activity).

[10]Note that comprehensive results of all eight Internet paths are available at https://confluence.slac.stanford.edu/display/IEPM/Decision+Theoretic+Approach.

TABLE IV
GOODNESS-OF-FIT RESULTS (MEASUREMENTS FROM SLAC)

(a) Gaussian distribution

Site	χ^2	p-value	C.Val@0.05
UTOR	14.18	0.2892	21.026
CERN	15.46	0.217	21.026
ORNL	18.22	0.1089	21.026
FZK	19.09	0.0864	21.026
SDSC	21.02	0.1009	23.685

(b) Weibull distribution

Site	χ^2	p-value	C.Val@0.05
DESY	17.30	0.1385	21.026
SWITCH	8.8	0.71	21.026
NSLABS	13.45	0.337	21.026

This can be summarized as follows:

$$\mathcal{H}_0 : \mathcal{R}_i = n \tag{5}$$
$$\mathcal{H}_1 : \mathcal{R}_i = n + m_i, \tag{6}$$

where n represents the random variable characterizing the baseline distribution of bandwidth estimates. As discussed earlier, the datasets under consideration are either Gaussian or Weibull distributed. Therefore, in the following sections we devise a model for datasets featuring Gaussian or Weibull distributions.

1) Decision-Theoretic Model for Gaussian Paths: For Gaussian distributed typical bandwidth measurements, let n in (5) and (6) represents a Gaussian random variable. For ease of exposition, we remove the first moment bias from n to make it a zero-mean normal distribution $n = \mathcal{N}(0, \sigma^2)$. We can represent m_i as: $m_i = \mathcal{R}_i - n$.

When a new bandwidth estimate arrives, it is mapped to one of the two hypotheses using the following conditional probability distributions:

$$\begin{aligned} \Pr(\mathcal{R}_i|\mathcal{H}_0) &= \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-\mathcal{R}_i^2}{2\sigma^2}\right); \text{ and} \\ \Pr(\mathcal{R}_i|\mathcal{H}_1) &= \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(\mathcal{R}_i - m_i)^2}{2\sigma^2}\right). \end{aligned} \tag{7}$$

A likelihood ratio test [40] to choose between the two hypotheses can then be defined as:

$$\Lambda(\mathcal{R}_i) = \frac{\Pr(\mathcal{R}_i|\mathcal{H}_1)}{\Pr(\mathcal{R}_i|\mathcal{H}_0)}. \tag{8}$$

Assuming independence between real-time bandwidth measurement, an aggregate likelihood for a set of measurements $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_N\}$ can be formulated as:

$$\Lambda(\mathcal{R}) = \prod_{i=1}^{N} \frac{\frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(\mathcal{R}_i - m_i)^2}{2\sigma^2}\right)}{\frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-\mathcal{R}_i^2}{2\sigma^2}\right)}. \tag{9}$$

Solving (9) using (5) and (6), we get:

$$\ln \eta \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\lessgtr}} \frac{1}{2\sigma^2} \sum_{i=1}^{N} \left(\mathcal{R}_i^2 - n^2\right), \tag{10}$$

where η is a tunable threshold parameter.

2) Decision-Theoretic Model for Weibull Paths: Similarly, if we let n represent a Weibull random variable and remove the mean bias, the two hypotheses can be written as:

$$\begin{aligned} \Pr(\mathcal{R}_i|\mathcal{H}_0) &= \left(\tfrac{k}{\lambda}\right)\left(\tfrac{\mathcal{R}_i}{\lambda}\right)^{(k-1)} \exp\left(-\left(\tfrac{\mathcal{R}_i}{\lambda}\right)^k\right); \text{ and} \\ \Pr(\mathcal{R}_i|\mathcal{H}_1) &= \left(\tfrac{k}{\lambda}\right)\left(\tfrac{\mathcal{R}_i - m_i}{\lambda}\right)^{(k-1)} \exp\left(-\left(\tfrac{\mathcal{R}_i - m_i}{\lambda}\right)^k\right), \end{aligned} \tag{11}$$

where $k, \lambda > 0$ represent the distribution's shape and scale parameters.

The likelihood ratio test to choose between the two hypotheses can then be defined as:

$$\Lambda(\mathcal{R}_i) = \prod_{i=1}^{N} \frac{\left(\tfrac{k}{\lambda}\right)\left(\tfrac{\mathcal{R}_i - m_i}{\lambda}\right)^{(k-1)} \exp\left(-\left(\tfrac{\mathcal{R}_i - m_i}{\lambda}\right)^k\right)}{\left(\tfrac{k}{\lambda}\right)\left(\tfrac{\mathcal{R}_i}{\lambda}\right)^{(k-1)} \exp\left(-\left(\tfrac{\mathcal{R}_i}{\lambda}\right)^k\right)}. \tag{12}$$

Solving (12) using (5) and (6) yields:

$$\ln \eta \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\lessgtr}} \sum_{i=1}^{N} \left((k-1)\ln\left(\frac{n}{\mathcal{R}_i}\right) + \left(\frac{\mathcal{R}_i - n}{\lambda}\right)^k\right). \tag{13}$$

For both Weibull and Gaussian distributions, n is the distribution of median-filtered baseline bandwidth values. As new bandwidth estimates arrive, they are plugged into the likelihood ratio defined in (10) or (13). The output of the test is then compared to an upper threshold η_1 and a lower threshold η_0. If $\Lambda(\mathcal{R}_i) \leq \eta_0$, we accept the null hypothesis $\mathcal{H}_0$. Alternatively, if $\Lambda(\mathcal{R}_i) \geq \eta_1$, we accept the alternate hypothesis $\mathcal{H}_1$. If neither case is true, we conclude that we do not have enough information to make a decision and wait for the next measurement to recalculate $\Lambda(\mathcal{R}_i)$.

D. Threshold Optimization

Wald showed [41] that we can define the thresholds η_0 and η_1 in terms of the rate of true-positive (or detection) rate p_D and the false-positive ratio p_F; note that the FP ratio here is different from our previous (/day) definition of FP rate; Wald defined p_F as the ratio of the false alarms and the total number of typical observations and henceforth we will refer to it as the *FP ratio*. It was shown that these rates may be approximated by user-defined values α and β such that:

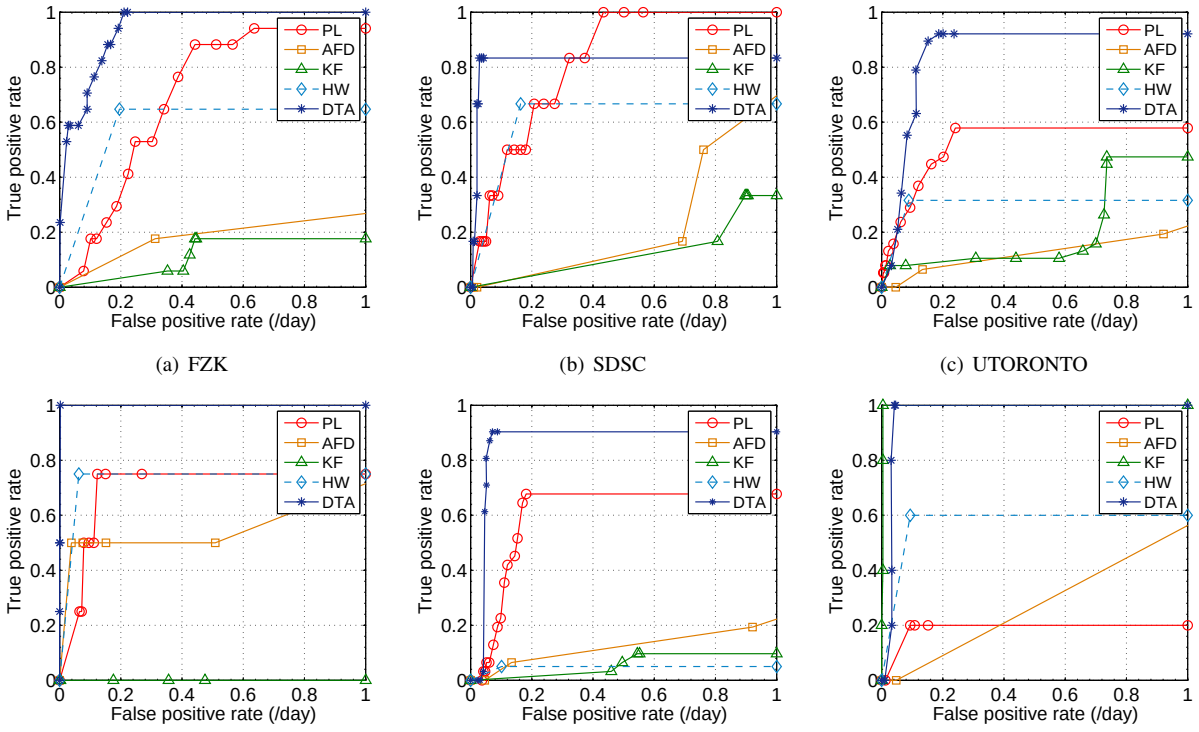
$$p_F \leq \alpha \qquad \text{and} \qquad p_D \geq \beta. \tag{14}$$

Through empirical evaluation, we conclude that TP rate p_D and FP ratio p_F defined in terms of $\alpha = 0.2$ and $\beta = 0.99$ yields acceptable results[11].

As an example, consider that the alternate hypothesis is accepted when it is in fact true; i.e. (10) met the threshold: $\eta_1 \leq \frac{\Pr(\mathcal{R}|\mathcal{H}_1)}{\Pr(\mathcal{R}|\mathcal{H}_0)}$. This means that the detection rate p_D is at least η_1 times the FP ratio p_F when $\mathcal{H}_1$ is true. Consequently, we can define η_1 and η_0 as:

$$\eta_1 \leq \frac{p_D}{p_F} \quad \text{and} \quad \frac{1 - p_D}{1 - p_F} \leq \eta_0. \tag{15}$$

[11]The range of values used to analyze the dataset and compile results presented in Section VI are posted at https://confluence.slac.stanford.edu/display/IEPM/Decision+Theoretic+Approach



(a) FZK (b) SDSC (c) UTORONTO

(d) NSLABS (e) DESY (f) SWITCH

Fig. 7. Accuracy comparison of the proposed Decision-Theoretic Approach (DTA) with Plateau Algorithm (PL), Adaptive Fault Detection (AFD), Kalman Filters (KF) method and the Holt-Winters (HW) method for pathChirp measurements; Figs. 7(a)–7(c) present results for Gaussian-distributed data sets while Figs. 7(d)–7(f) present results for Weibull-distributed data sets. Note that due to lack of space we present six ROC curves here; ROC curves of all eight Internet paths are available at https://confluence.slac.stanford.edu/display/IEPM/Decision+Theoretic+Approach.

Using the bounds defined in (14), we get:

$$\eta_1 = \frac{\beta}{\alpha} \quad \text{and} \quad \eta_0 = \frac{1-\beta}{1-\alpha}. \tag{16}$$

While the above upper and lower thresholds satisfy the user-defined detection and false-positive constraints, from experiments we observed that the upper threshold of η_1 renders the algorithm too sensitive to data sets with large variances. In order to allow the algorithm to adapt itself to data sets in which the variance of the bandwidth measurements changes over time, we redefine the upper threshold η_1 as:

$$\eta_1 = \frac{\beta\sigma^2}{\alpha}, \tag{17}$$

where σ^2 is the variance of the median-filtered data being analyzed for anomalies.

Step-wise execution of this decision-theoretic approach to real-time path event detection is outlined in Algorithm 2.

VI. Accuracy and Detection Delay of the Proposed Approach

A comparison of the proposed Decision-Theoretic Approach (DTA) with the AFD, KF, HW and PL algorithms is shown in Fig. 7. It is clear that the proposed approach provides consistently and considerably higher accuracy than all the existing methods. AFD and KF methods provide significantly lower detection accuracy than the proposed DTA because of the reasons enumerated in Section IV-C1. The only exception is Fig. 7(f) where KF performs better than the proposed DTA algorithm. This is because the events on this path are consecutive (i.e., anomalous bandwidth measurements occur in a burst) and KF, which is designed to capture recent measurements, finds it easy to detect these bursty events. Plateau algorithm has a much higher FP rate than DTA because it operates on rigid thresholds given as input to the algorithm. Consequently, if an Internet path changes its characteristics even slightly (e.g., increases its variance due to changes in cross-traffic,) Plateau is unable to adapt its parameters in accordance with the slightly modified baseline behavior. Holt-Winters also performs relatively poorly since it tries to model noise as seasonal trends.

In case of Fig. 7(b), 7(c) and 7(e) we observe that, while having considerably higher accuracy than existing algorithms, the proposed DTA algorithm does not achieve 100% detection rate. This is because the median filter length was chosen to be 15. The selection of this length implies that any anomaly existing for less than 4 hours (i.e., 8 measurements) will be filtered and thus will not be detected. This limitation can be addressed by reducing the median filter length. A direct consequences of this selection will be an increase in sensitivity and the FP rate. We see in Table V that in case of SDSC one event was not detected; upon further investigation we observed that the duration of this undetected event was three hours due to which it was removed by the median filter and not considered as an anomaly. We therefore conclude

Algorithm 2: Event detection.

Data: Array of performance measurements Δ, false positive rate α, detection rate β, window size ρ, initial duration for training dataset τ and width of median filter ν.
Result: Array of timestamp-brackets Ψ classifying windows as containing events.

Apply low-pass median filter of width ν to obtain Δ_{tr};
Compute μ_{tr} and σ_{tr} for $\{\delta_t \in \Delta_{tr} | t_0 < t < t_0 + \tau\}$;
Let threshold $\eta_1 = \frac{\beta\sigma^2}{\alpha}$, $\eta_0 = \frac{1-\beta}{1-\alpha}$ and $t_0 = 0$;
`/* determine the baseline */`
Apply χ^2 test on Δ_{tr} to determine baseline distribution;
`/* initialize the observation window */`
Let $\tau_s = t_0 + \tau - \nu$ and $\tau_e = t_0 + \tau$;
for $\{\omega \in \Omega\}$ **do**
    Let $x_1 = rand()$, $x_2 = rand()$ and $n = \sqrt{-2\ln(x_1)} \cdot sin(2\pi x_2) \cdot \sigma$;
    $R = median\{\omega_i | \tau_s \leq i \leq \tau_e\}$;
    Compute η ;
    if $\eta_1 < \eta$ **then**
        Observation δ is anomalous, add $\delta's$ timestamp to the array of events Ψ;
    else if $\eta < \eta_0$ **then**
        Observation δ is not anomalous;
        Update the training dataset with δ, discard the oldest entry, recalculate σ_{tr} and η_1;
    else Not enough information to make a decision;
    Increment τ_s and τ_e;
end
Analyze Ψ and combine consecutive anomalous windows defining unique events;

that selecting a median filter of lesser length increases the detector's sensitivity at the cost of slightly higher FP rates.

In summary, KF, HW and AFD achieve 100% detection rates at very high false-positive rates. PL which performs better than KF, HW and AFD with acceptable levels of detection-rates and false-positive rates is superseded by DTA with higher detection-rates for acceptable levels of false-positive rates.

Detection delays of DTA are provided in Table V. Comparison with Table III shows that DTA incurs nearly the same detection delays as that of PL, HW and KF. DTA performs slightly better than PL on four links, but slightly worse on the other four. Same is the case with HW. On the other hand, the KF method features lower delays, but it does so with poor TP rates. The AFD method presents the worst results in comparison to other methods. We therefore conclude that DTA, while providing considerably higher accuracy, does not introduce a significant increase in detection delays.

VII. Conclusions and Future Work

In this paper, we first evaluated the accuracies and detection delays of existing anomaly detectors for event detection over end-to-end Internet paths. We observed that most existing anomaly detectors and network metrics are unsuitable for anomaly detection beyond enterprise boundaries. Existing algorithms that can be adapted for the present problem provide acceptable detection delays, but their accuracies are quite low. We then revealed statistical characteristics of Internet bandwidth measurements that can facilitate detection. Based on these characteristics, we proposed a decision-theoretic anomaly detector (DTA). We demonstrated the application of the DTA approach to Gaussian and Weibull distributed Internet bandwidth measurement and showed that DTA can achieve considerably higher accuracy than existing detectors while having similar detection delays.

As part of our ongoing work, we are developing a tool based on the proposed DTA approach. This tool and a wrapper for the `RRDtool` will be released as open-source and will replace the Plateau algorithm at the SLAC National Accelerator Laboratory's Internet measurement infrastructure.

TABLE V
Average Detection Delay $\bar{t}$ of the Proposed DTA Approach (in Terms of Number of Additional Observations Required Before an Event is Detected)

	Decision Theoretic Approach				
	Detected		*Undetected*		*Total*
	#	$\bar{t}$	#	$\bar{t}$	
UTOR	35	11.29	3	∞	38
CERN	8	13.53	0	-	8
DESY	30	5.29	1	∞	31
SDSC	5	6.60	1	∞	6
FZK	17	8.60	0	-	17
TRIUMF	2	3.70	1	∞	3
NSLABS	4	0.00	0	-	4
SWITCH	5	0.60	0	-	5

References

[1] "iPerf." [Online]. Available: http://dast.nlanr.net/projects/iperf/
[2] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions," in *ACM SIGCOMM*, 2005.
[3] T. Ahmed and M. Coates, "Multivariate Online Anomaly Detection Using Kernel Recursive Least Squares," in *IEEE INFOCOM*, 2007.
[4] P. Barford and J. Kline, "A Signal Analysis of Network Traffic Anomalies," in *IMC - ACM SIGCOMM*, 2002.
[5] H. Ringberg, A. Soule, and C. Diot, "Sensitivity of PCA for traffic anomaly detection," in *SIGMETRICS*, 2007.
[6] T. D. Lane, "Machine Learning Techniques for the computer security domain of anomaly detection," Ph.D. dissertation, Department of Electrical and Computer Engineering, Purdue University, 2000.
[7] L. Cottrell, "Internet End-to-end Performance Monitoring - Bandwidth to the World (IEPM-BW) project," 2002. [Online]. Available: http://confluence.slac.stanford.edu/display/IEPM
[8] W. Matthews and L. Cottrell, "The PingER project: active Internet performance monitoring for the HENP community," in *IEEE Communications Magazine*, 2000.
[9] P. Calyam and A. Kalash, "RICE: A Reliable and Efficient Remote Instrumentation Collaboration Environment," in *IMMERSCOM*, 2007.
[10] C. H. Huang, "BioGrid: a collaborative environment for Life Science Research," in *A.P.I.C.E.* Springer Milan, 2006.
[11] D. Sisalem and A. Wolisz, "Towards TCP-Friendly Adaptive Multimedia Applications Based on RTP," in *IEEE Symposium on Computers and Communications*, 1999.
[12] M. Claypool and A. Tripathi, "Adaptive Video Streaming using Content-Aware Media Scaling." [Online]. Available: http://citeseer.ist.psu.edu/claypool04adaptive.html
[13] C. Logg and L. Cottrell, "Experiences in traceroute and available bandwidth change analysis," in *ACM SIGCOMM workshop on Network Troubleshooting*, 2004, pp. 247–252.
[14] H. Hajji, "Statistical Analysis of Network Traffic for Adaptive Faults Detection," in *IEEE Transactions on Neural Networks*, 2005.
[15] A. Soule and el al., "Combining Filtering and Statistical Methods for Anomaly Detection," in *IMC - ACM SIGCOMM*, 2005.
[16] J. D. Brutlag, "Aberrant Behavior Detection in Time Series for Network Monitoring," in *USENIX LISA XIV*, 2000.
[17] NIST/SEMATECH, *e-Handbook of Statistical Methods, 6.4.3.5. Triple Exponential Smoothing*, 2003. [Online]. Available: http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc435.htm

[18] IPerf, "The TCP/UDP Bandwidth Measurement Tool." [Online]. Available: http://dast.nlanr.net/Projects/Iperf/

[19] pathChirp, "Bandwidth Estimation Tool." [Online]. Available: http://www.spin.rice.edu/Software/pathChirp/

[20] S. Shalunov, "thrulay: Network Capacity and Delay Tester." [Online]. Available: http://shlang.com/thrulay/

[21] N. Merhav, M. Gutman, and J. Ziv, "On the estimation of the order of a Markov chain and universal data compression," in *IEEE Transactions on Information Theory*. IEEE, 1989, pp. 1014–1019.

[22] S. Shalunov, B. Teitelbaum, A. Karp, J. Boote, and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)," *Network Working Group - RFC 4656*, September 2006, standards Track.

[23] M. Jain and C. Dovrolis, "End-to-End Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput," in *ACM/IEEE Transactions on Networking*, August 2003, pp. 537–549.

[24] "IEPM Data Set." [Online]. Available: https://confluence.slac.stanford.edu/display/IEPM/Home

[25] "PingER participants." [Online]. Available: http://pinger.fnal.gov/participants.html

[26] L. Cottrell and et al., "Evaluation of Techniques to Detect Significant Network Performance Problems using End-to-End Active Network Measurements," in *NOMS*, 2006.

[27] A. Shriram and et al., "Comparison of Public End-to-End Bandwidth Estimation Tools on High Speed Links," in *PAM*, 2005.

[28] A. Shriram and J. Kaur, "Empirical Evaluation of Techniques for Measuring Available Bandwidth," in *IEEE INFOCOM*, 2007.

[29] A. Lakhina, K. Papagiannaki, and C. Diot, "Structural Analysis of Network Traffic Flows," in *ACM SIGMETRICS*, 2004.

[30] F. Hussain, U. Kalim, N. Latif, and S. A. Khayam, "A Performance Evaluation of Anomaly Detection Algorithms for Internet Paths," Tech. Rep., 2009. [Online]. Available: https://confluence.slac.stanford.edu/download/attachments/22086003/tr-event-detection.pdf?version=1

[31] Y. Gu and et al., "Detecting anomalies in network traffic using maximum entropy estimation," in *IMC - ACM SIGCOMM*, 2005.

[32] A. Lakhina, M. Crovella, and C. Diot, "Characterization of network-wide anomalies in traffic flows," in *IMC - ACM SIGCOMM*, 2004.

[33] M. Williamson, "Throttling viruses: restricting propagation to defeat malicious mobile code," in *IEEE ACSAC*, 2002.

[34] J. Jung and et al., "Fast portscan detection using sequential hypothesis testing," in *IEEE Symposium on Security and Privacy*, 2004.

[35] S. S. Jaeyeon and et al., "Fast Detection of Scanning Worm Infections," in *RAID*, 2004.

[36] M. V. Mahoney, "Network Traffic Anomaly Detection Based on Packet Bytes," in *ACM SAC*, 2003.

[37] M. V. Mahoney and P. K. Chan, "PHAD: Packet Header Anomaly Detection for Identifying Hostile Network Traffic," Tech. Rep., 2001.

[38] T. Fawcett, "An Introduction to ROC Analysis," *Pattern Recognition Letters, Science Direct*, 2006.

[39] D. Heeger, "Signal Detection Theory," *Dept. of Psychology, NYU*, 2007. [Online]. Available: http://www.cns.nyu.edu/~david/handouts/sdt/sdt.html

[40] H. V. Trees, *Detection, Estimation, and Modulation Theory, Part I*. John Wiley and Sons, 2001. [Online]. Available: http://gunston.gmu.edu/demt/demtp1/

[41] A. Wald, *Sequential Analysis*. John Wiley and Sons, 2004.