J Netw Syst Manage (2014) 22:208–234
DOI 10.1007/s10922-013-9286-0

Topology-Aware Correlated Network Anomaly Event Detection and Diagnosis

Prasad Calyam · Manojprasadh Dhanapalan · Mukundan Sridharan · Ashok Krishnamurthy · Rajiv Ramnath

Received: 21 November 2012 / Revised: 1 September 2013 / Accepted: 16 September 2013 /
Published online: 26 September 2013

Abstract For purposes such as end-to-end monitoring, capacity planning, and performance bottleneck troubleshooting across multi-domain networks, there is an increasing trend to deploy interoperable measurement frameworks such as perfSONAR. These deployments expose vast data archives of current and historic measurements, which can be queried using web services. Analysis of these measurements using effective schemes to detect and diagnose anomaly events is vital since it allows for verifying if network behavior meets expectations. In addition, it allows for proactive notification of bottlenecks that may be affecting a large number of users. In this paper, we describe our novel topology-aware scheme that can be integrated into perfSONAR deployments for detection and diagnosis of network-

This material is based upon work supported by the Department of Energy under Award Numbers: DE-SC0001331 and DE-SC0007531. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

P. Calyam (✉)
University of Missouri-Columbia, Columbia, MO, USA
e-mail: calyamp@missouri.edu

M. Dhanapalan · R. Ramnath
The Ohio State University, Columbus, OH, USA
e-mail: dhanapalan.1@osu.edu

R. Ramnath
e-mail: ramnath.6@osu.edu

M. Sridharan
The Samraksh Company, Dublin, OH, USA
e-mail: mukundan.sridharan@samraksh.com

A. Krishnamurthy
RENCI, Chapel Hill, NC, USA
e-mail: ashok@renci.org

wide correlated anomaly events. Our scheme involves spatial and temporal analyses on combined topology and uncorrelated anomaly events information for detection of correlated anomaly events. Subsequently, a set of ‘filters’ are applied on the detected events to prioritize them based on potential severity, and to drill-down upon the events “nature” (e.g., event burstiness) and “root-location(s)” (e.g., edge or core location affinity). To validate our scheme, we use traceroute information and one-way delay measurements collected over 3 months between the various U.S. Department of Energy national lab network locations, published via perfSONAR web services. Further, using real-world case studies, we show how our scheme can provide helpful insights for detection, visualization and diagnosis of correlated network anomaly events, and can ultimately save time, effort, and costs spent on network management.

Keywords Performance monitoring · Correlated anomaly detection · Spatial and temporal analysis filters · perfSONAR measurement framework · Active measurements

1 Introduction

The pace of scientific discovery has been rapid in recent years owing to cyberinfrastructures that enable researchers to: (a) remotely access distributed computing resources and big data sets in clouds, and (b) effectively collaborate with remote peers, at a global-scale. Given the fact that remote access and collaboration often occur over networks that are multi-domain, there is a need to instrument individual network domains with interoperable measurement frameworks that enable them to join a multi-domain measurement federation. The motivation for joining a measurement federation is to facilitate measurements across multiple domains over the Internet for reaping the mutual benefits of performing ‘end-to-end’ network monitoring, capacity planning, and performance bottleneck troubleshooting.

To serve the needs of multi-domain measurements, there is a rapidly growing trend to deploy interoperable measurement frameworks such as perfSONAR [1] that assist in measurement data collection, storage and publishing/subscribing of data archives of current and historic measurements (e.g., bandwidth, delay, jitter and loss) via web services. The web service schemas in perfSONAR have been standardized in the Open Grid Forum [2] and have been adopted in frameworks such as OSCARS [3], Cricket-SNMP [4], and PingER [5]. Using web services, any site can register with the global lookup service [6] to allow intra-domain or inter-domain users to initiate measurements from their measurement servers or to query their published measurement datasets. Numerous perfSONAR deployments are sampling both active and passive measurements of various metrics several times a day. They are exposing these collected measurements via web services in the form of vast data archives of current and historic measurements on national and international backbones (e.g., ESnet, Internet2, GEANT, SWITCH). The successful adoption of perfSONAR can be attributed to the open, modular, and distributed architecture.

The consumers of the perfSONAR active measurements (e.g., network operators and researchers in data-intensive science disciplines) are now faced with the challenge to analyze, visualize, and interpret the vast measurement data sets across end-to-end multi-domain network paths with minimal human inspection. They direly need automated techniques and intuitive tools to query, analyze, detect, diagnose, and notify prominent network performance anomaly events such as plateaus within active measurements that may hinder data transfer speeds [7, 8]. Timely and accurate bottleneck anomaly event detection coupled with effective notification can lead to quicker resolution of network faults, and thus can proactively prevent large numbers of end-users from experiencing annoyingly slow data transfers or poor interaction response times of critical applications on well-provisioned high-speed networks. In addition, anomaly event notifications allow for verifying whether network behavior meets expectations, especially when known application traffic events such as daily data backups or scheduled network maintenances occur. Further, anomaly event notifications can suggest improved performance plateaus for network operators to validate their network capacity upgrades.

Uncorrelated network anomaly events (change-points from statistical norm) can be detected at the ‘network-path level’ by analyzing for e.g., end-to-end one-way delay and throughput measurement time series from OWAMP and BWCTL active measurement tools used in perfSONAR deployments, respectively. In comparison, correlated network anomaly events can be detected at the ‘network-wide level’ by analyzing several network-path level (uncorrelated) anomaly events in order to *localize* the change-cause to a particular network segment. There have been several schemes developed to automatically and accurately detect uncorrelated network anomaly events [7–9] within multi-domain active measurements that are publicly-accessible through frameworks such as perfSONAR. However, in this same context, there is a clear dearth of schemes in prior literature that leverage topology information along with a measurement time series of network health metrics to address the challenges of automated detection and drill-down of correlated network anomaly events. Existing works such as [10, 11] use network domain-specific operations information such as router logs and command histories, which are typically not readily available in perfSONAR-like multi-domain measurement infrastructures, and hence are not applicable. The ability to detect and diagnose correlated network anomaly events using topology and a measurement time series can be helpful in multi-domain measurement infrastructures to: (a) group uncorrelated events as having a common root-cause(s), (b) order anomaly events based on *severity* (i.e., the entry with the highest ratio value of ‘common event ratio temporally’ to ‘common hops ratio spatially’ tops the list) to help efficient isolation and remediation of bottlenecks, and (c) narrow down the diagnosis to determine the pertinent location(s) of the common root-cause(s).

In this paper, we address the above dearth and present our novel ‘network topology-aware correlated anomaly detection and diagnosis’ (NTA-CAD) scheme that can be integrated into perfSONAR deployments. Our scheme involves spatial and temporal analyses on combined topology and uncorrelated anomaly events information from a measurement time series for detection of correlated anomaly events. Subsequently, a set of filters is applied on the detected events to prioritize them based on severity and to drill-down upon the events nature (e.g., event burstiness) and root-cause location(s) (e.g., edge or core location affinity). The severity prioritization analysis involves matrix manipulations to classify correlated anomaly events into grids of 'Low–Low' to 'High–High' along axes of spatial commonness (i.e., "common hop ratio") and of temporal commonness (i.e.,"common event ratio"). To validate our scheme, we use traceroute information and one-way latency measurements collected over 3 months between the various U.S. Department of Energy (DOE) national lab network locations, published via perfSONAR web services. Further, using real-world case studies, we show how our scheme can provide helpful insights for detection and diagnosis of correlated network anomaly events. For example, we illustrate how to identify paths that are critical temporally (e.g., paths with links that have high common event frequency) and spatially (e.g., paths that have links that are common amongst most of the critical paths). To accomplish these analyses, we leverage representations that are used typically in studies involving social networks [12] and apply them to the context of active network measurement data analysis. Finally, our scheme implementation in this paper builds upon perfSONAR web service interfaces and is designed to be integrated into widely-used anomaly notification dashboards such as RACF perfSONAR dashboard [13] and can thus save time, effort, and costs spent on network management in high-speed networking communities.

The remainder of the paper is organized as follows. Section 2 discusses related work. Section 3 describes the problem scope and motivations for our work. Section 4 presents the novel approaches for data transformations in our NTA-CAD scheme. Section 5 details the network-wide results and filters evaluation with our NTA-CAD scheme in a DOE labs perfSONAR data set case study. Section 6 concludes the paper.

2 Related Work

To assist network operators in troubleshooting bottlenecks (e.g., prolonged congestion events or device mis-configurations) in high-speed networks, a number of smart and effective network monitoring tools based on statistical measurement data analysis techniques have been developed in earlier literature. In particular, there have been studies on correlated anomaly detection such as [14–16]. Principal component analysis (PCA) discussed in [14] focuses on network anomaly detection on a network link basis. In [16], the authors aim to overcome limitations of PCA's failure in detecting strong correlations in distributed network traffic anomalies. Both [14] and [16] do not use topology information and can be considered as black-box techniques in comparison to other aware works such as [10, 15, 17]. The authors in [15] use Kalman-filter for anomaly detection and build a traffic matrix of an enterprise network to overcome link basis limitations. In [10], the authors present a general framework called Network-wide Information Correlation and Exploration

(NICE) for analyzing data through correlations and present a qualitative as well as quantitative analysis approach with network related data such as router logs and topology information. Routing connection relationships are used in [17] for network-wide anomaly detection in backbone networks; relationships are established based on features such as packet sizes, IP addresses and ports.

Our NTA-CAD work is closely related to the NICE framework [10] and can even be integrated into their framework to perform detailed diagnosis and troubleshooting of active measurements that are not originally supported. NICE uses a database repository that houses diverse passive sets of data such as router syslogs, work-flow data, and routing protocol events. Such a data repository is built to support correlated anomaly detection schemes and can be leveraged in our NTA-CAD scheme analysis for verification with potential ground truths. Works such as [18–20] use correlation analysis over different measurement data sets. However, their motivation is to optimize the measurement frequency, whereas our scheme works on infrastructures involving multi-domain networks whose measurement sampling frequency is typically selected by network domain administrators. In [11], the authors use a campus-specific network anomaly event data set to classify anomalies and describe a workflow for dealing with an anomaly event after it has been detected.

There have been a few studies such as [21] that have focused on developing methods for classifying anomalies for prioritization purposes. The authors in [21] define a set of attributes and conditions based on ports and packets source/destination fields for classifying anomalies. A signature based data analysis is discussed in [22], where historical events are analyzed to extract signatures that are subsequently indexed for efficiently matching events to a current sample signature. Works such as [23] use basic visualizations of perfSONAR time-series data for manual detection of correlated anomaly events. Our NTA-CAD scheme would be best suited to augment these visualizations for automated detection of correlated anomaly events in perfSONAR data sets, and our novel concepts such as applying spatial and temporal filters combined with drill down can be effective for prioritizing, identifying root-cause locations, and the nature of anomaly events for bottleneck resolution.

3 Problem Scope and Motivation

3.1 Anomaly Event Detection

One of the significant challenges in dealing with measurement data sets is to decide which kind of network events need to be labeled and notified as anomaly events and finding the cause of a given anomaly event. Various traffic related anomaly events are caused due to IP route/AS path change events that involve traffic re-routing on backup paths due to ISP traffic migration for maintenance reasons involving BGP policy modifications [14, 24] or handling cable faults [25], cyber attack events involving malware/worms [26] and botnets [27], as well as misconfigurations in router advertisements [8] and access router bandwidth settings [28]. These events manifest in the form of spikes, dips, bursts, persistent variations and plateau trends in network performance metrics such as end-to-end round-trip delay, available bandwidth and loss obtained through end-to-end active measurements. Some anomaly events that manifest as short intermittent spikes, intermittent dips, and bursts in network performance metrics are not of interest for notification as anomaly events because they are generally caused due to user-behavior during normal network operations i.e., users generating various application traffic.

Based on experiences from network operators and application users [8, 28] and based on our extensive discussions with other network operators supporting data-intensive science and engineering communities (e.g., ESnet, Internet2, GEANT), the notification of ‘plateau anomalies’ shown in Fig. 1 are the most worthy to be notified. These anomaly events are commonly known to impact data transfer speeds at the application-level on high-speed network paths. Network operators, when analyzing a measurement time-series of network performance metrics, typically look for plateau event trends (such as the example event shown in Fig. 1) through visual inspections and seek for automated notification of such network-wide detected anomaly events from measurement systems. Variants of plateau anomaly event detectors have been developed and adopted in large scale monitoring infrastructures such as NLANR AMP [7] and SLAC IEPM-BW [8], which are predecessors to the perfSONAR deployments. These detectors detect that a plateau event or a ‘change event’ has occurred if the most recent measurement sample value crosses the upper or lower thresholds of the summary (i.e., T_{SU}, T_{SL}) and quarantine (i.e., T_{QU}, T_{QL}) buffers that are continuously updated over time as new samples arrive. The summary buffer is used to maintain sample history that indicates the normal state (before anomaly event occurs), and a quarantine buffer is used to store outlier data samples that are twice the normal state sample values. The sample counts in these buffers are used to maintain trigger count values over a pre-configured trigger duration before an alarm of anomaly event occurrence (indicated by the cross mark in Fig. 1) is notified. The trigger duration before samples are marked for impending anomaly states (triangle symbols shown in Fig. 1) should be chosen long enough to avoid false alarms due to noise events corresponding to intermittent spikes, dips, or bursts [9].

The main limitation in the plateau detectors in NLANR AMP and SLAC IEPM-BW deployments is that they used static configurations of salient threshold parameters to detect change-points from the statistical norm. Specifically, they used a static configuration of “sensitivity” and “trigger elevation” threshold parameters for increasing the probability of anomaly event detection while at the same time decreasing the probability of false alarms. The sensitivity parameter is used to specify the magnitude of plateau change that may result when an anomaly event on a network path is to be triggered for notification. The trigger elevation parameters are used to temporarily increase the thresholds to avoid repeated triggers for a brief period after an anomaly event has been detected. The sensitivity and trigger elevation threshold parameters in static plateau detection (SPD) schemes need to be manually calibrated to accurately detect plateaus in different measurement sample profiles on network paths. Such a laborious process for accurate anomaly detection

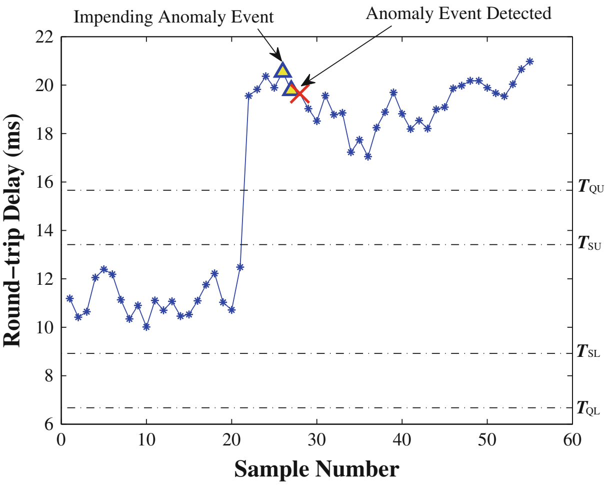


Fig. 1 Plateau event with upper and lower thresholds illustration

is impractical for large-scale monitoring infrastructures involving large numbers of end-to-end network paths with dynamically changing traffic characteristics.

To overcome the shortcomings of such SPD schemes, we recently developed an adaptive plateau detection (APD) scheme [9] for perfSONAR deployments. The APD dynamically configures thresholds to accurately detect ‘uncorrelated’ network anomaly events. The intuition behind the dynamic configuration is based on reinforcement learning principles and the observation that raw measurements just after an anomaly event provide direct intelligence about the anomaly event itself; leveraging them for reinforcement of the machine learning (to compare statistics of historic and current measurement samples for detecting change points) can make the uncorrelated anomaly detection more robust and accurate. The difference in the working of APD in comparison to SPD can be understood as follows. For any example trace given, SPD schemes commonly choose a static sensitivity setting of 2, which is known to produce relatively low false alarms compared to other settings through empirical observations of similar traces. For the same case, APD dynamically updates the sensitivity values during trace analysis as new samples are considered over time and produces lower false alarm rates at the cost of a fractional increase in detection time that is needed for the reinforcement learning.

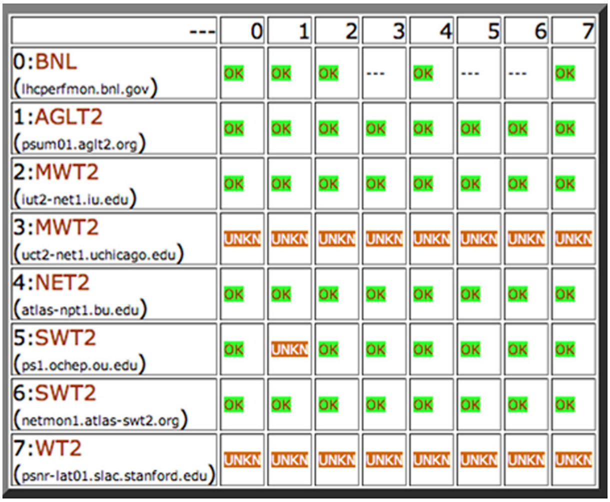
A reliable plateau detector to trigger uncorrelated network anomaly events is a critical component for notifying correlated network anomaly events, and for this reason we adopt the APD to detect uncorrelated anomaly events, which we subsequently analyze to find correlations in these events with our NTA-CAD scheme presented in this paper.

3.2 Anomaly Event Notification

Our APD scheme implementation is currently deployed in network monitoring environments of perfSONAR communities such as the Energy Sciences Network

(ESnet), DOE E-Center [29] and US Atlas [13]. The anomaly event notification in these communities is being done using Nagios plugins [30] of APD and visualized in the form of dashboards (e.g., RACF perfSONAR dashboard) [13] as shown in Fig. 2. These dashboards are helpful to alert network-path level anomaly events using status messages such as: ‘OK’, which is marked in green to indicate that the network path does not have any current anomaly events, and ‘CRITICAL’, which is marked in red to indicate that the network path is currently experiencing anomaly events. In our interactions with these communities of network operators and data-intensive science researchers, we realized that additional context for diagnosis and localization was needed when ‘CRITICAL’ alerts were found i.e., information about alert severity, event characteristics and the network segment that caused the anomaly event was needed to effectively troubleshoot any bottlenecks.

To illustrate why additional context is needed for effective troubleshooting, let us consider 3 sites say BNL, AGLT2 and MWT2 and their related paths BNL-AGLT2, BNL-WT2, and AGLT2-WT2 that are shown in Fig. 2. Let us also assume that there are network-path level (i.e., uncorrelated) anomaly events detected on these paths over a specific time period, and we have the timestamps of when the anomaly events were detected. With only this information, it is challenging and time consuming for network operators to diagnose the event root-cause and location. Now, as the number of paths to be monitored increases, and/or the number of anomaly events detected over a specific time period increases, the challenge and time consumption for the network operators increases drastically. However, based on topology information, let us assume it is determined that the paths BNL-AGLT2, BNL-MWT2, and AGLT2-WT2 share many common links that are likely the root-cause for the triggered anomaly events (*spatial analysis*). Also, based on inter-event times information, let us assume that it is determined that the events detected on the path BNL-WT2 occurred in a relatively more bursty manner within a short time interval



---	0	1	2	3	4	5	6	7
0:BNL (lhcperfmon.bnl.gov)	OK	OK	OK	---	OK	---	---	OK
1:AGLT2 (psum01.aglt2.org)	OK	OK	OK	OK	OK	OK	OK	OK
2:MWT2 (iut2-net1.iu.edu)	OK	OK	OK	OK	OK	OK	OK	OK
3:MWT2 (uct2-net1.uchicago.edu)	UNKN	UNKN	UNKN	UNKN	UNKN	UNKN	UNKN	UNKN
4:NET2 (atlas-npt1.bu.edu)	OK	OK	OK	OK	OK	OK	OK	OK
5:SWT2 (ps1.ochep.ou.edu)	OK	UNKN	OK	OK	OK	OK	OK	OK
6:SWT2 (netmon1.atlas-swt2.org)	OK	OK	OK	OK	OK	OK	OK	OK
7:WT2 (psnr-lat01.slac.stanford.edu)	UNKN	UNKN	UNKN	UNKN	UNKN	UNKN	UNKN	UNKN

Fig. 2 RACF perfSONAR dashboard using APD on OWAMP measurements

(*temporal analysis*). Now, we can combine these 2 analyses' results to inform the network operator about which anomaly event has a higher severity in terms of performance degradation that needs prioritized attention (*combined analysis and drill down*). By eliminating any bottlenecks (e.g., device mis-configurations or inadequate bandwidth provisioning) to resolve the higher priority and more severe anomaly event on BNL-WT2, it may likely result in the resolution of the other two detected anomaly events on BNL-AGLT2 and AGLT2-WT2, hence saving time, effort, and costs spent by the network operator on network management.

4 Network Topology Aware Approach

4.1 Overview

Figure 3 shows the various system components that support our NTA-CAD analysis scheme. In the first step (Step 1: 'input step'), active measurement time-series data (i.e., temporal data) are downloaded from perfSONAR archives through the openly accessible web service interfaces as explained in Sect. 4.2 Once the temporal data sets are downloaded, they are then analyzed in the second step (Step 2: 'matrices manipulation step') along with network topology data (i.e., spatial data) downloaded from topology services maintained by network operators deploying perfSONAR measurement points. The analysis is performed through matrices manipulation on a network path level basis for (uncorrelated) event detection using our APD scheme [9] as explained in Sect. 4.3 If the number of paths being analyzed is large or if the data volumes are large due to high sampling frequency of measurements or due to measurement data queries being made over large time periods, the APD analysis can be done in a distributed manner at various measurement point locations, and the event information can be relayed to the Network Operations Center location for further analysis and bottleneck resolution. It is obviously possible to also have the Network Operations Center be the only location for all of the APD as well as the NTA-CAD analysis to be performed. We showed in [9] that in such scenarios, the agility of analysis can be improved by using parallel queries of web services, versus using sequential queries. In addition, it should be noted that the effectiveness of the NTA-CAD analysis is dependent on the reliability of the topology information (especially in the cases of paths with many common links), and is affected if there is lack of information such as missing links [31] or misinformation such as spurious third-party addresses [32] or lack of up-to-date intermediate link capacity information. Finally, as part of the third step (Step 3: 'output step'), joint analysis with the detected uncorrelated anomaly events information in the NTA-CAD analysis is performed to prioritize the events based on correlation characteristics such as event location or event burstiness as explained in Sect. 4.4 The prioritized events information can simultaneously be visualized through pertinent highlighting in dashboards monitored by the Network Operations Center personnel.

Algorithm 1 NTA-CAD Scheme Steps

```
Require: List of Paths
/*[Step 1:] Data Collection and Processing*/
for all Paths do
    Collect time series data of active measurements (e.g., OWAMP, BWCTL)
    Collect traceroute data
end for
/*[Step 2:] Event Detection and Analysis*/
for all Pathpairs i, j do
    $E_c[i,j]$ = Number of common events between $i$ and $j$
    $E_t[i,j]$ = Total number of events in both $i$ and $j$
    $H_c[i,j]$ = Number of common links between $i$ and $j$
    $H_t[i,j]$ = Total number of links in both $i$ and $j$
end for
for all $H_c[i,j]$ do
    $H_n[i,j] \leftarrow \frac{2*H_c[i,j]}{H_t[i,j]}$ /* Find common hop ratio between paths*/
end for
for all $E_c[i,j]$ do
    $E_n[i,j] \leftarrow \frac{2*E_c[i,j]}{E_t[i,j]}$ /* Find common event ratio between paths*/
end for
$E_{n'} \leftarrow E_c \geq t_e$ /* Select common hop matrix elements with value greater than $t_e$*/
$H_{n'} \leftarrow H_c \geq t_h$ /* Select common event matrix elements with value greater than $t_h$*/
/*[Step 3:] Output Presentation and Drill Down*/
$EH_{n'} \leftarrow (E_{n'}).(H_{n'})$
$PriorityList <> \leftarrow Grid(EH_{n'})$
DrillSpace(PriorityList<>)
DrillTime(PriorityList<>)
```

In the following, we explain the details of the 3-step sequence involved in the NTA-CAD scheme for correlated anomaly detection and diagnosis shown in Fig. 4; we also refer to Algorithm 1 that outlines the various computation details within the 3-step sequence.

4.2 Data Collection and Processing

The data collection involves querying distributed measurement archives (accessible at an address e.g., http://testproject.example-university.edu:8085) by using perfSONAR-compliant web service clients. The site list of measurement archives that are available for query can be selected using a global lookup service [6] hosted by the perfSONAR community. This service registers the addresses of all openly-accessible measurement archives within individual domains. Through standardized request/response messages, active measurement time series data relating to end-to-end performance measurement tools such as BWCTL (TCP/UDP throughput) and OWAMP (one-way delay as specified in IETF RFC 4656) as well as traceroute data are downloaded. The downloaded time series data in the form of XML files are then processed using parsing for further analysis.

4.3 Event Detection and Analysis

For anomaly detection in the processed data, we use our APD scheme that performs uncorrelated path level anomaly detection. The time-series data sets that have been annotated to have anomaly events by an APD are examined for further common-hop (spatial) and common-event (temporal) analysis. In the common-hop analysis that

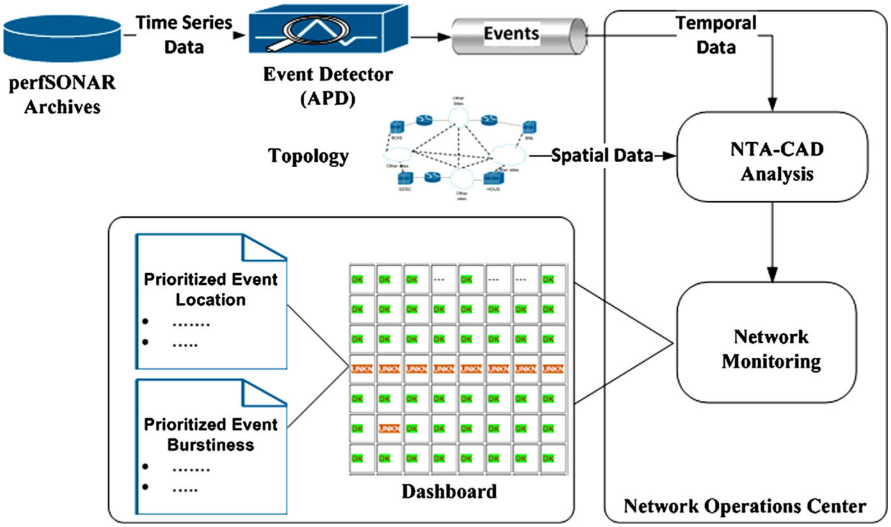


Fig. 3 System components that support the NTA-CAD scheme

leverages the network-wide route topology data obtained from the traceroute tool, we analyze every pair of network-wide paths using a simple common hop scheme. We look into all the intermediate links in a path and compare it with another path to find out the number of common links that these paths share between them. A path-to-path matrix is constructed for representing common links between the paths, which we call the *common hops matrix* H_c. For example: $H_c[A, B] = 8$ implies 8 common links exist in the A and B paths. Another matrix H_t contains the sum of total links in the two paths. For comparison purposes, we normalize the matrix for further use and consequently create another matrix called H_n, the normalized common hops matrix, where each element is calculated as a ratio as shown in Eq. 1 for two paths say, A and B.

$$\text{Element in } H_n = \frac{(2 \ * \ Number\ of\ common\ links\ between\ A\ and\ B)}{(Total\ number\ of\ link\ \sin\ both\ A\ and\ B)} \tag{1}$$

In the common-event analysis, we study the anomaly events information of network-wide paths being monitored with the APD, and determine those anomaly events that are happening around the same time window. To better understand how the study is done, we look at an event in say, path A at a time t and look for an event that happened around the same time within a certain time window in path B. If such common events between paths exist in time, we use that information to build a path-to-path matrix E_c and normalize it as a common event matrix E_n, where each element of this matrix is calculated as a ratio as shown in Eq. 2 for two paths say, A′ and B′.

$$\text{Element in } E_n = \frac{(2 \ * \ Number\ of\ common\ events\ between\ A'\ and\ B')}{(Total\ number\ of\ event\ in\ both\ A'\ and\ B')} \tag{2}$$

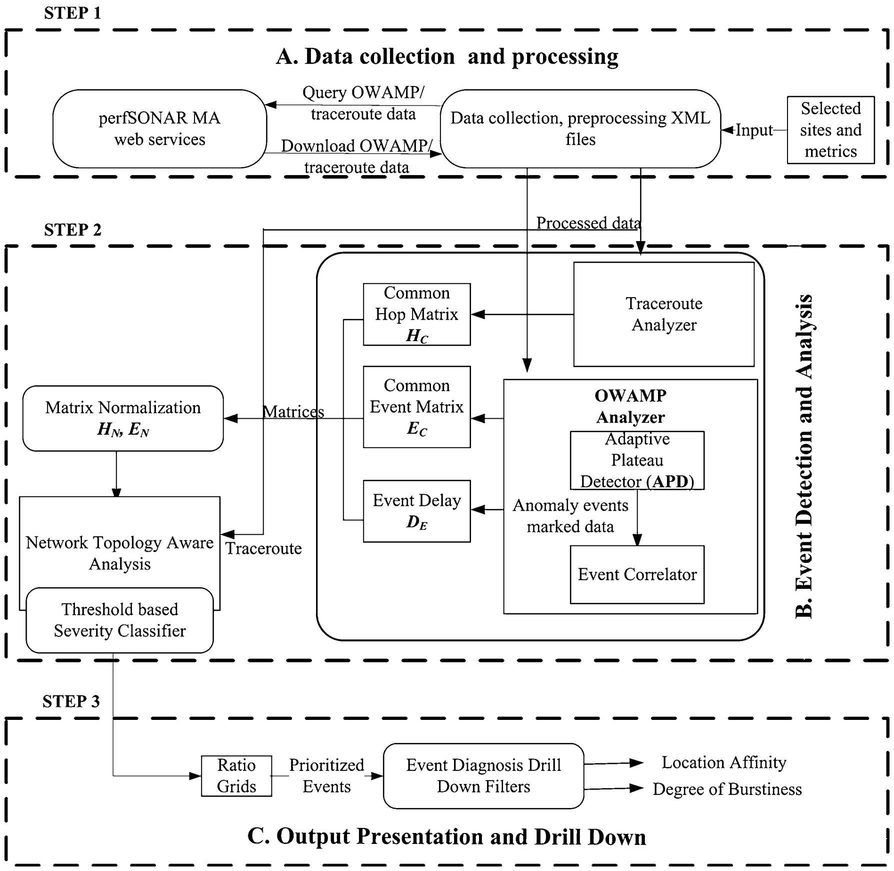

STEP 1
A. Data collection and processing
perfSONAR MA web services
Query OWAMP/ traceroute data
Download OWAMP/ traceroute data
Data collection, preprocessing XML files
Input
Selected sites and metrics
Processed data
STEP 2
Common Hop Matrix H_C
Traceroute Analyzer
Common Event Matrix E_C
OWAMP Analyzer
Adaptive Plateau Detector (APD)
Matrix Normalization H_N, E_N
Matrices
Event Delay D_E
Anomaly events marked data
Network Topology Aware Analysis
Traceroute
Event Correlator
Threshold based Severity Classifier
B. Event Detection and Analysis
STEP 3
Ratio Grids
Prioritized Events
Event Diagnosis Drill Down Filters
Location Affinity
Degree of Burstiness
C. Output Presentation and Drill Down

Fig. 4 NTA-CAD scheme block diagram

4.4 Output Presentation and Drill Down

As we reach this step, we have both H_n and E_n matrices that contain spatial and temporal information about the anomaly events occurring network-wide. We can now jointly perform correlated analysis and filter events of higher priority based on topology awareness. First, we filter only those events that have common links by applying spatial constraints on the H_n matrix. To illustrate this, consider a condition ($H_n > 0$) on the normalized common hop matrix. In such a condition, the resulting binary matrix $H_{n'}$ will contain only the paths that have at least one common link between them. Now applying this spatial constraint on the common event matrix, we have $E_n.(H_{n'})$ that would give us only those paths that share common links. Next we can filter with a $E_n > 0.5$ threshold condition to obtain $E_{n'}$ that represents a common event binary matrix with only those paths pairs that have at least half of the events happening within the same time window. Now to separate those path pairs with at least 50 % of events happening within the same time window and having at least one common link between them, a dot product of $(E_{n'}).(H_{n'})$ can be performed to obtain the resulting binary event matrix ($EH_{n'}$). The ($EH_{n'}$) matrix can now be utilized to overcome the current limitations in network monitoring dashboards that only indicate occurrence of uncorrelated anomaly events at a path-level. Network-wide correlated anomaly events based on common links and events can be filtered by configuring suitable thresholds for the normalized H_n and E_n matrices. The filtered anomaly events can be classified under different severity levels and plotted as a graph of 'common hops' versus 'common events' for each of the rows of the ($EH_{n'}$) matrix (on a normalized scale from 0 to 1), thus producing a more effective output presentation for any given analysis window. The thresholds can be selected by network operators in a custom fashion, or they can be selected qualitatively by dividing the severity space into equal sized grids using high (H), low (L) and medium (M) thresholds. Each of the anomaly events thus falls into one of the ($H|M|L$, $H|M|L$) grids that have a corresponding severity level. For example, anomaly events in the (H, H) grid are most severe and are of high priority for further investigation and resolution because they represent a network-wide state where a large number of anomaly events are occurring at the same time on paths that share many common links. Obviously, any common events in time between paths that do not share any common links between them may call for a different analysis perspective than those anomaly events that share several common links and happen close to each other in a time window. In any case, such an output presentation of correlated anomaly events provides network operators a more guided direction for proceeding with further diagnosis of the anomaly events using a spatial drill down (i.e., identify links that may be the root-cause) and a temporal drill down (i.e., analyze the inter-event times that can be quantified in terms of burstiness). In the following, we use an example scenario shown in Fig. 5 to illustrate how we can do a spatial and temporal drill down of anomaly events for diagnosis.

4.4.1 Spatial Drill Down

Let us consider the set of paths B, G and R shown in Fig. 5a that share common network segments among them. Due to this sharing, they feature in the common hop

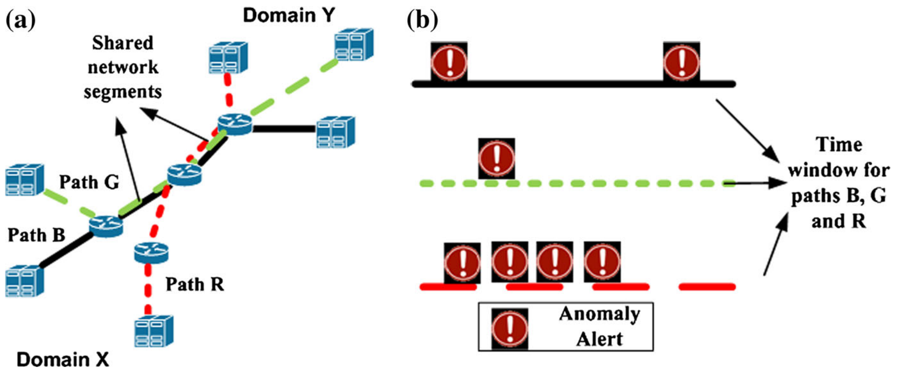


Fig. 5 **a** Paths B, G and R showing sharing of common network segments between them; **b** anomaly events occurrence for paths B, G and R

matrix as having a certain spatial relation between each other. When anomaly events occur on all these paths around the same time window, this information is captured in the common event matrix. In order to determine the anomaly event occurrence root-location(s), we can analyze the shared network segments or underlying common links using the spatial drill down. Paths B and G share between them a greater part of the common network segments than path R, and hence are relatively more interesting. Also, there is a small network segment closer to the edge of Domain Y, which is common to all the three paths experiencing anomaly events, and in our case could most likely have the links that have high "location affinity" in the core towards the edge of Domain Y i.e., this segment could contain the root-location causing the correlated anomaly events.

4.4.2 Temporal Drill Down

A temporal drill down can be further helpful in understanding the nature of the correlated anomaly events. Figure 5b shows different inter-event time spacing between anomaly events detected at different times on paths B, G and R. From our previous discussion on the spatial drill down, we know that path R only has a small segment that is common with the other paths B and G. However, from Fig. 5b we can see that the path R exhibits "burstiness" i.e., it has many anomaly events that occur within short intervals. Path G has just one event and path B has events that are relatively far apart. By such a temporal drill down, burstiness of paths in terms of anomaly events can be calculated, and the paths with higher burstiness can be investigated further with any other possible ground truth information (e.g., router logs). For quantifying burstiness, let us consider an analysis window within which we have anomaly event occurrence times obtained after APD processing. We can define burstiness B^* as shown in Eq. 3.

$$B^* = \frac{E_t}{E_T} \tag{3}$$

Note that E_t is the number of anomaly events within the analysis window with inter-event times below a threshold t_{time}, and E_T is the total number of anomaly events within the analysis window. As the numerator E_t increases, the value of burstiness B^* increases. Since burstiness characterizes anomaly events occurring in an analysis window on a normalized 0–1 scale, it does not provide information about the number of anomaly events occurring. Hence, we also look at the number of anomaly events within this analysis window and assign a weight or importance level to the burstiness B^* levels.

5 NTA-CAD Network-Wide Evaluation and Results

In this section, we validate our NTA-CAD scheme with a case study involving analysis of a perfSONAR data set collected between various geographically-distributed locations. We also illustrate how different spatial and temporal filters can be applied on the data set for the drill down and diagnosis of correlated network anomaly events.

5.1 Dataset

For our NTA-CAD scheme validation experiments, we use traceroute information and one-way delay measurements collected via perfSONAR web services over 3 months among the various U.S. Department of Energy (DOE) national lab network locations. More specifically, our data collection covers 17 DOE national lab sites shown in Fig. 6 selected from across the United States that have perfSONAR deployments. Our data query involves 216 paths and their performance data for approximately 50 days over a 3 month time span. We selected one-way delay measurement data from the OWAMP tool for our validation experiments because they contain more samples per day and have more detailed information of network status. We remark that the perfSONAR data set corresponding to the DOE national lab network locations is valuable since the DOE community has invested significant resources for training and deployment of perfSONAR. In addition, the paths whose performance data we sampled features traffic to-and-from large high-performance computing systems serving sites all over the world and supports a good mix of bandwidth-intensive and latency-sensitive application cross traffic.

5.2 Common Hop and Event Analysis

Figure 7 shows our traceroute data analysis to obtain a histogram of the common hops distribution over the entire topology. We can see that there are several path pairs with one common hop, and the number of path pairs decreases significantly as the number of common hop count increases. There were path pairs we found that had as much as 9 common hops between them. We remark that we do not show the zero common hops bin, however we found 41010 path pairs within the total 46656 path pairs analyzed over 216 paths that had no common hops between them. Thus, we can see that using spatial filters based on common hops relationship on topology

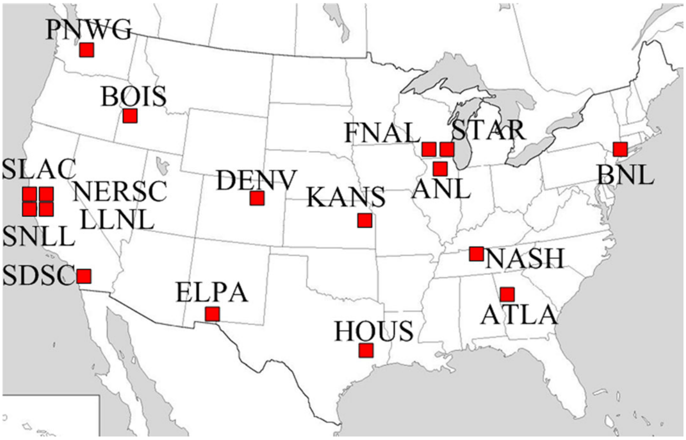


Fig. 6 US Map showing various DOE national lab network locations

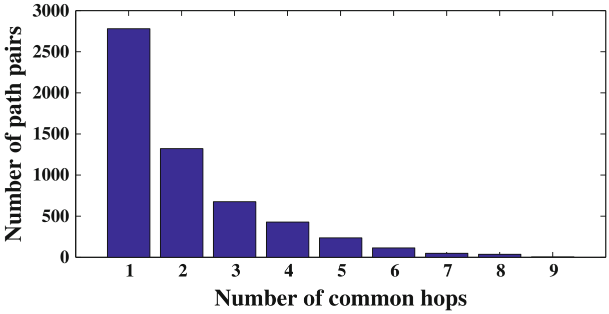


Fig. 7 Common hops distribution over entire topology (Note: Outlier value for path pairs sharing zero common hops is 41,010 within the total 46,656 path pairs analyzed over 216 paths)

information can greatly reduce the data processing to identify critical links and paths in troubleshooting correlated anomaly events.

To further analyze the spatial relationships between the different DOE lab sites in terms of common hops that may have significant impact on network-wide path pairs, we use the Graphviz [33] tool, which is an open source graph visualization software that is commonly used to visualize social network graphs. The Graphviz tool output is shown in Fig. 8, where graph nodes represent path pairs of the normalized path-to-path matrix H_n of the topology, and the edges represent the common hop ratio strength values between the paths. The matrix H_n could be trimmed at various levels using different threshold configurations. For example, we can look at clustering of the path pairs which have at least 0.65 common hop ratio values. We can see that some paths have more edges than other paths, and the paths with the most number of edges can be assumed to have a greater impact on the network status. Thus, we can identify spatially critical paths in this case study that are annotated in Fig. 8, whose performance issues need to be more carefully monitored than other paths.

A similar analysis can be performed on the temporal relationships between the different DOE lab sites in terms of common events in the one-way delay measurement data set that may have a significant impact on network-wide path pairs. Figure 9 shows the graph nodes that represent path pairs of the normalized path-to-path matrix E_n of the detected anomaly events, and the edges represent the common event ratio strength values between the paths. The matrix E_n could also be trimmed at various levels using different threshold configurations. For example, we can look at clustering of the path pairs that have at least 0.70 common event ratio values. We can see that some paths have more edges than other paths, and the paths with the most number of edges can be assumed to have a greater impact on the network status. Thus, in this case study, we can identify the most critical path (STAR-BNL) annotated in Fig. 9 whose performance issues need to be more carefully monitored than other paths.

5.3 Ratio Grids

In this section, we analyze the anomaly events within the case study data set and classify them into one of the (*H*|*M*|*L*, *H*|*M*|*L*) grids that has a corresponding severity

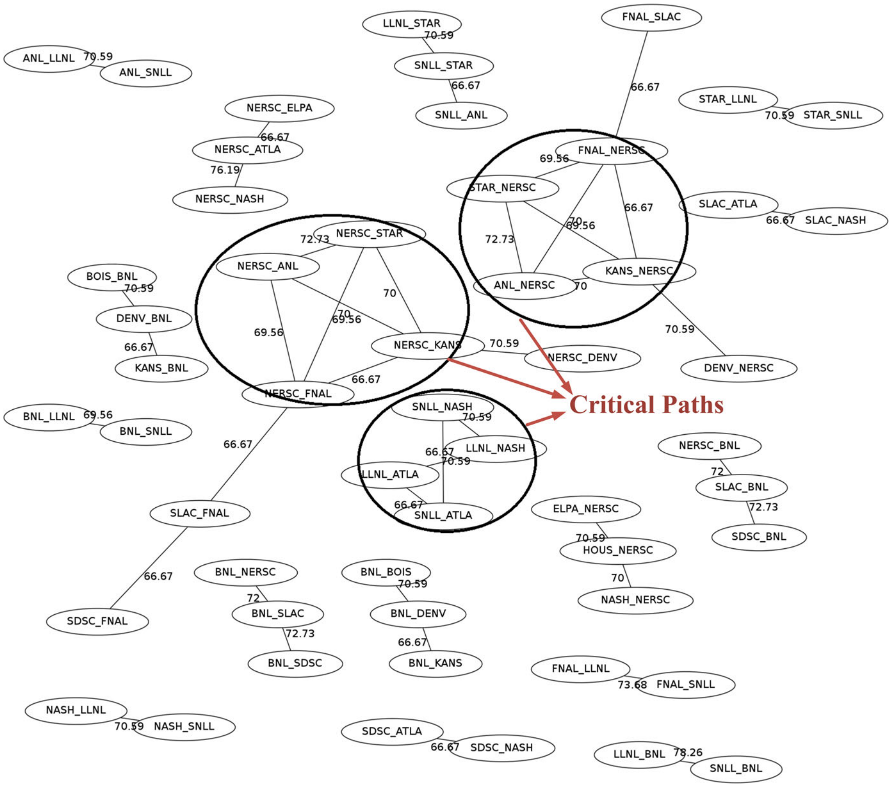


Fig. 8 Common hop graph showing paths having a common hop ratio above >0.65

level. To construct the grid, any one path can be chosen as the reference path; we choose the FNAL-BNL as the reference path for illustrative purposes. As mentioned earlier, we select thresholds for differentiating the grids qualitatively by dividing the severity space within the ($EH_{n'}$) matrix into equal sized grids with a common hop ratio as y-axis and common event ratio as x-axis. All the points on the x-axis represent those events between paths that do not have any common hops. Likewise, events on the y-axis represent those events that do not have any common events. The points falling within the (L, L) grid have very low spatial and temporal significance and can be given less priority compared to the points falling within the (H, H) grid that have the highest spatial and temporal significance.

Figures 10 and 11 show the grids plotted with events from the FNAL-BNL reference path correlated with all other paths in the network when SPD and APD are applied on the case study data set, respectively. Note that the events shown in the case of APD correspond to data spanning *over 3 months*, whereas the events in the case of SPD correspond to data spanning only for *2 weeks*. From these Figures., we can make two major observations. First, the classification of anomaly events within the grids can be helpful to network operators to quickly analyze network-wide status and also prioritize the troubleshooting of most significant anomaly events. Second,

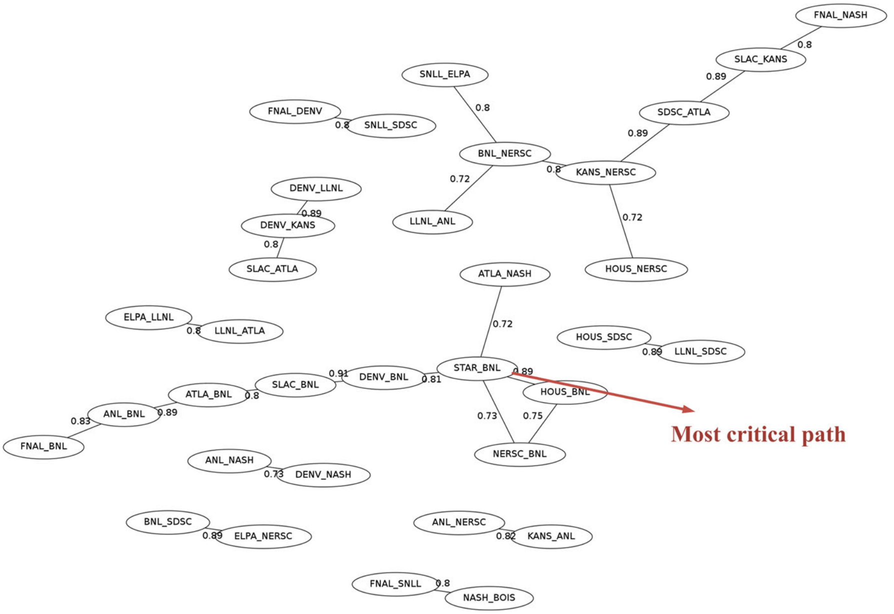


Fig. 9 Common events graph showing paths having a common event ratio above >0.70

we can realize that the APD plot has much less noise than the SPD over an extended period of data analysis, and the SPD can be extremely dense and noisy in terms of false alarms, even over short periods of data analysis. We remark that we found a 10–15 % higher number of alarms of plateau events reported by the SPD in comparison to the APD in our data sets. Details of improved anomaly detection accuracy without false positives or false negatives of the APD over the SPD can be found in [9] for a variety of traces with time series characteristics that include events such as persistent decrease, persistent increase, intermittent bursts, intermittent dips, and persistent variations.

5.4 Spatial Drill Down Filter

In addition to analyzing critical paths and monitoring high priority anomaly events on these paths, we can also drill down the topology data to identify the most common links across these critical paths. Links that are part of most critical paths can be given a high ‘path score’ and a ranking list can be generated to carefully monitor these links in comparison to other links. For our case study data set, Table 1 shows such a ranking list for the links among the paths shown in Fig. 9 with a high common event ratio between them.

To illustrate the application of a spatial drill down filter in our case study data set, we select a set of paths {FNAL-ATLA, KANS-FNAL, SDSC-FNAL, SNLL-FNAL} within the case study data set that are comprised of links with high path scores, and also are part of critical paths with high common event ratios. Figure 12

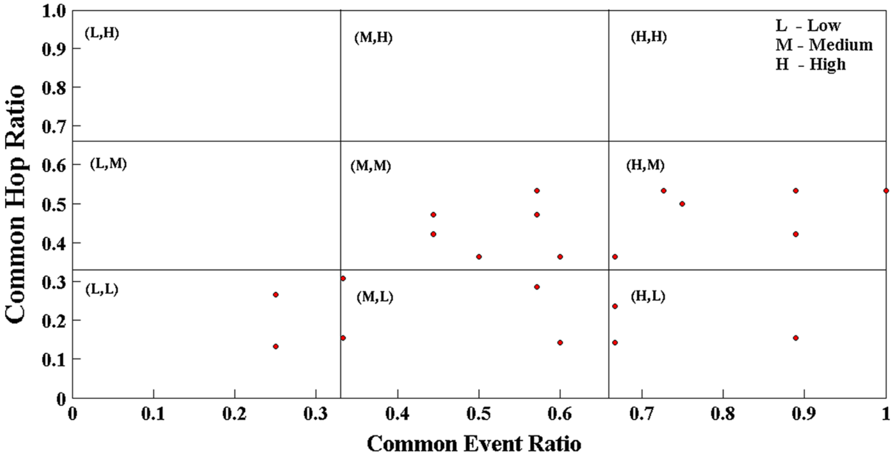


Fig. 10 APD grid showing events between FNAL_BNL and all other paths over 3 months

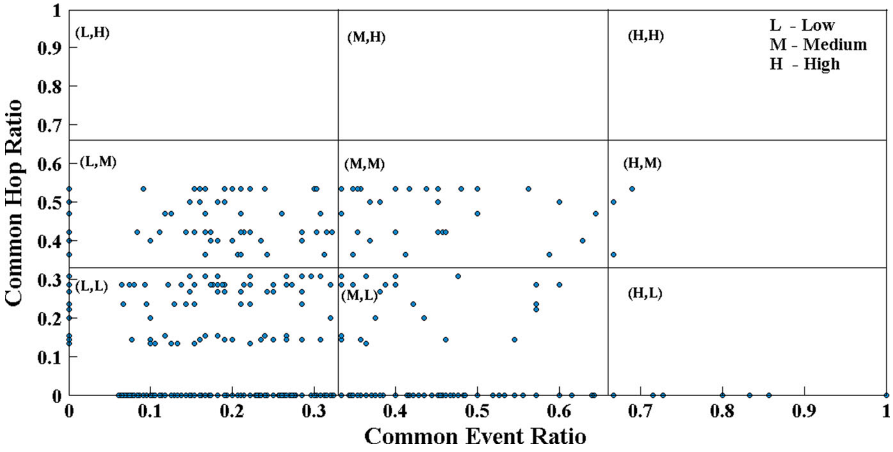


Fig. 11 SPD grid showing noisy events between FNAL_BNL and all other paths over 2 weeks

shows the links on these paths with correlated events. We can see that there are many links that have only one path in common, however there are other links that have up to 4 paths in common, and all the links that have more than one path in common are closer to the FNAL edge. Hence, we can conclude that some factor closer to the FNAL edges and may be causing several of the correlated anomaly events seen on all the select set of paths. Obviously, such a conclusion can be used as a guidance, and additional information sources (e.g., router logs, maintenance activity logs) need to be referred to the network segments closer to the FNAL edge to determine the ground truth. In any case, such a guidance and additional context of spatial filter analysis is extremely helpful to network operators when there are several network-wide events with varying degrees of severity showing up as ‘red’

Table 1 Links ranked based on path score

Links	Number of paths
R5-134.55.221.58–R6-134.55.209.46	11
R14-134.55.217.141–R15-198.124.238.49	9
R16-134.55.209.97–R17-134.55.217.1	9
R9-134.55.209.46–R10-134.55.220.49	9
R18-134.55.41.145–R19-134.55.41.121	8
R20-134.55.217.53–R18-134.55.41.145	8
R19-134.55.41.121–R14-134.55.217.141	8
R10-134.55.220.49–R16-134.55.209.97	7
R17-134.55.217.1–R21-134.55.219.9	6
R21-134.55.219.9–R22-134.55.217.41	6
R22-134.55.217.41–R23-134.55.217.33	6
R23-134.55.217.33–R24-198.129.254.33	6
R25-134.55.220.37–R26-198.124.252.97	5
R27-198.129.252.45–R9-134.55.209.46	5
R28-134.55.38.185–R16-134.55.209.97	5
R20-134.55.217.53–R5-134.55.221.58	5
R28-134.55.220.149–R29-134.55.38.185	5
R29-134.55.217.153–R30-198.129.254.141	5

alerts on their dashboards. Note that the link 134.55.221.58–134.55.209.46 in Fig. 12 has 2 paths in common in the context of events being analyzed to be affected near to the FNAL edge. However, its path score overall in the network is 11 as indicated in Table 1, and hence resolution of any problem corresponding to this link is of relatively high importance from a network-wide perspective.

5.5 Temporal Drill Down Filter

In addition to analyzing critical paths and monitoring high priority anomaly events on these paths, we can also drill down the instantaneous measurement data to identify the event burstiness and related manifestations in the critical paths. Recall that event burstiness as described in Sect. 4.4 is based upon inter-event time spacing between anomaly events detected and can help in determining how rapidly the anomaly events are occurring. We analyzed around 5462 inter-event time spacing samples within the case study data set and found the mean of these inter-event times (μ) to be 2,912 s (≈48 min). We remark that the sampling frequency of the measurements within the individual domains (i.e., individual DOE lab domains in our case study) is a major factor that influences the mean calculation. Figure 13 shows the network-wide event burstiness results for the case study data over 50 day samples (i.e., samples are collected on a daily basis in our data set for 50 days over a 3 month time span) with different inter-event time thresholds. The thresholds are chosen based on a geometric progression involving fractions of μ in the range of $[\mu/2, \mu/4, \mu/6, \ldots, \mu/14]$ on the order of minutes. We can see that the average

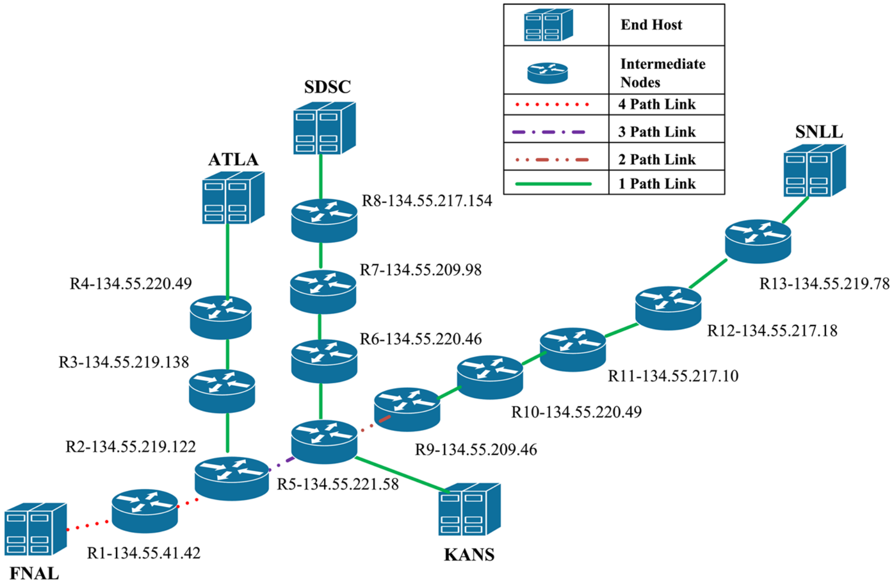


Fig. 12 Locations of all the common links originating from FNAL

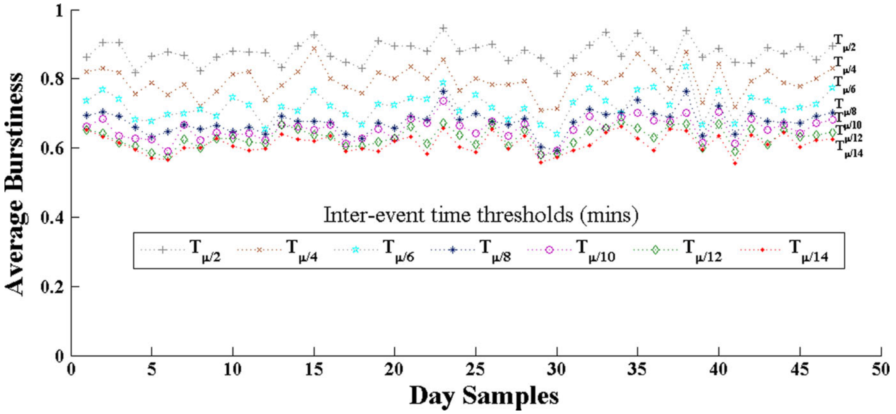
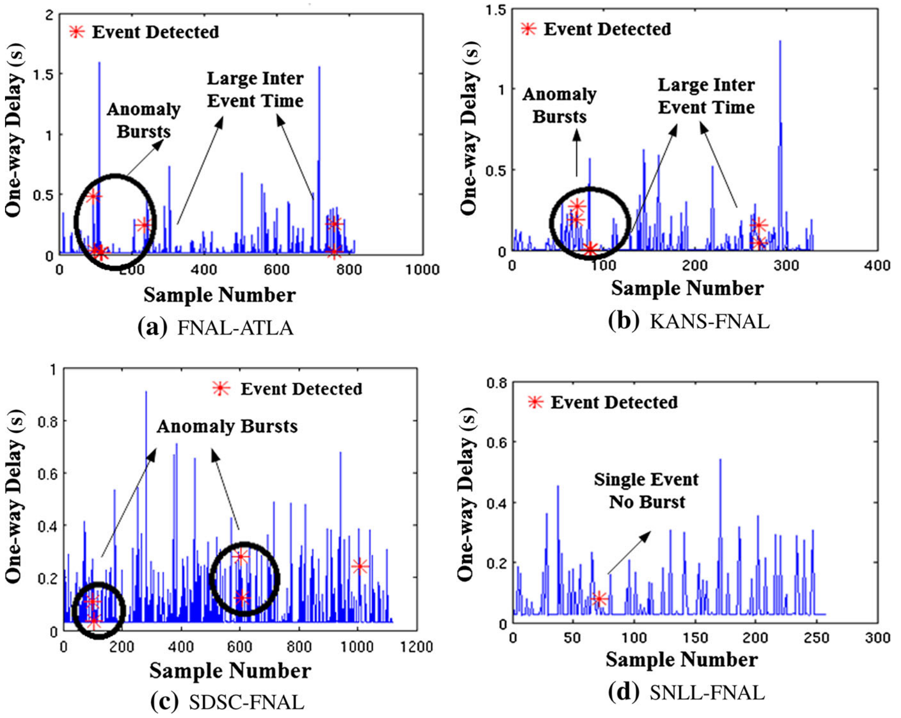


Fig. 13 Effect of various inter-event time thresholds on Burstiness

burstiness at the granularity of any given day decreases with a decrease in the inter-event time thresholds, and burstiness becomes steady at a lower range of threshold values. The day granularity of average burstiness network-wide is a reasonable timescale for network operators for receiving alerts and potential resolution of bottlenecks.

To illustrate the application of a temporal drill down filter in our case study data set, we select the same set of paths {FNAL-ATLA, KANS-FNAL, SDSC-FNAL, SNLL-FNAL} within the case study data set that are comprised of links with high path scores, and also are part of critical paths with high common event ratios.



(a) FNAL-ATLA

(b) KANS-FNAL

(c) SDSC-FNAL

(d) SNLL-FNAL

Fig. 14 Burstiness of APD events in one-way delay measurements

Table 2 Burstiness results for case study paths

Paths	Burstiness	Number of events
FNAL-ATLA	0.5	7
KANS-FNAL	0.6	6
SDSC-FNAL	0.5	5
SNLL-FNAL	0.0	1

Figure 14 shows the anomaly events annotated instantaneous performance graphs of one-way delay measurements for these four paths. Based on our burstiness definition in Sect. 4.4, we can apply a burstiness based temporal filter on the common events data and obtain Table 2 that shows the burstiness values along with the corresponding number of common events. Obviously, the path SNLL-FNAL that just has one event has zero burstiness. Among the other paths, we can observe that the FNAL-ATLA and SDSC-FNAL have the same burstiness value. However, the FNAL-ATLA has 7 common events, which is higher than the 6 and 5 for the KANS-FNAL and SDSC-FNAL, respectively. Thus, although burstiness value gives a quantifiable measure of how quickly anomaly events are occuring, the number of common events information supplements the burstiness information in prioritization of critical paths for troubleshooting, and the paths with relatively higher burstiness and common events need to have a higher priority during troubleshooting.

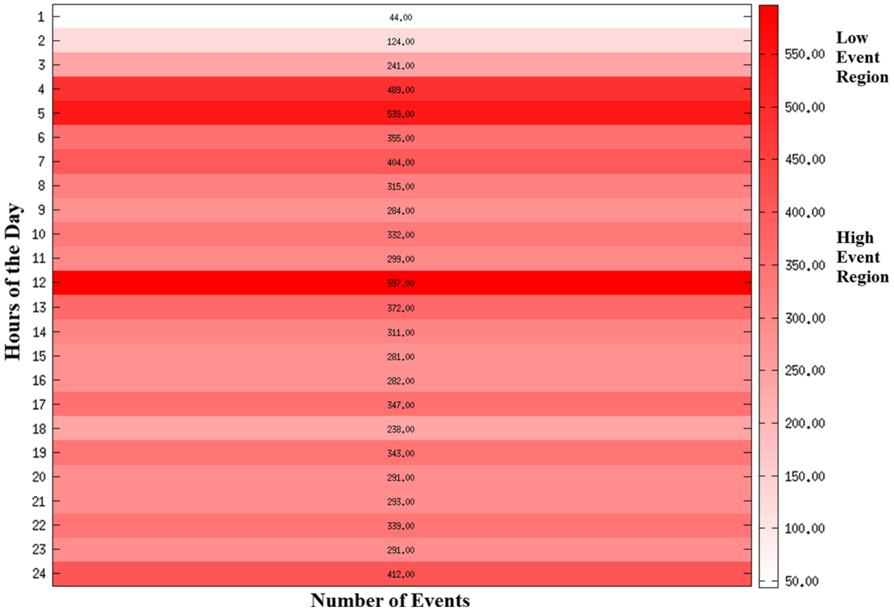


Fig. 15 Distribution of network-wide events on a per-hour basis over the case study data set

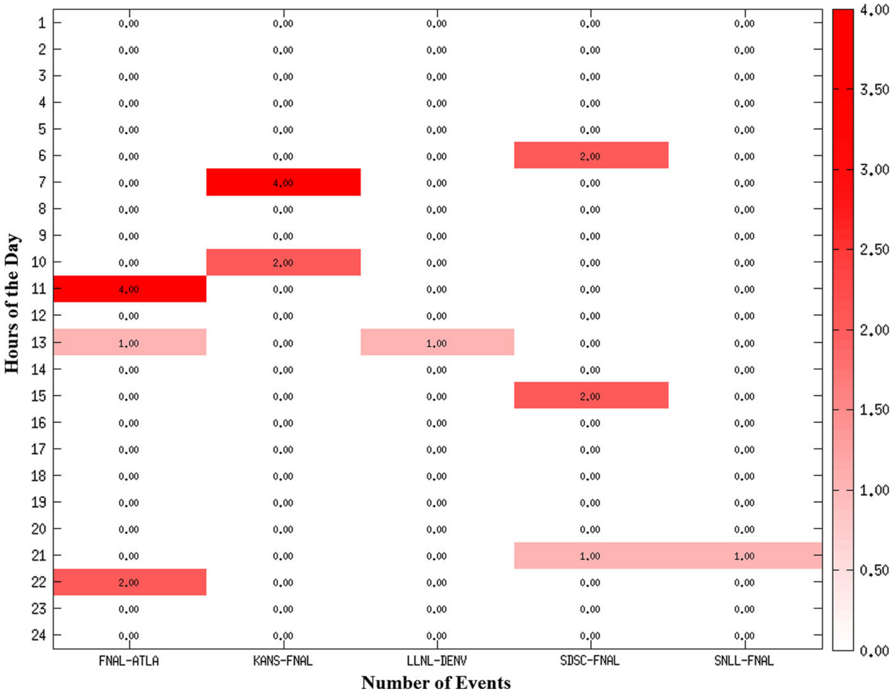


Fig. 16 Distribution of anomaly events of critical paths on a per-hour basis

Several other temporal drill down filters can be applied to the case study data set to obtain interesting guidance regarding the temporal nature of the anomaly events. We looked at a history based temporal filter that can be helpful in expectation management of network status. Figure 15 shows the heat map visualization of network-wide events distribution on a per-hour basis for the entire case study data set. We can easily identify the times during the day such as between 12:00 and 13:00 h that experience a higher number of anomaly events. Breaking down the event distribution for the critical links as shown in Fig. 16, we can identify which paths contribute the most for the heat level of the network-wide status in the heat map visualization. Obviously, impact of any periodic events that are part of the expectation of network status, such as scheduled daily backups can be easily identified with this temporal filter, and those can be ignored in the context of troubleshooting.

6 Conclusion and Future Work

In this paper, we presented a novel topology-aware scheme that can be integrated into perfSONAR monitoring dashboards for detection and diagnosis of network-wide correlated anomaly events across multiple domains. Our scheme involves using an adaptive plateau detector (APD) to generate anomaly events with low false alarms, and applying spatial and temporal analyses by combined topology information to detect correlated anomaly events. We devised a set of ‘filters’ that can subsequently be applied on the detected correlated anomaly events to prioritize them into one of the (*H*|*M*|*L*, *H*|*M*|*L*) (i.e., High, Medium, Low) grids based on potential severity. We showed how spatial and temporal drill-down of the events can reveal information relating to the “nature” (e.g., event burstiness) and “root-location(s)” (e.g., edge or core location affinity). Such information provides helpful guidance in identifying clusters of critical hops/links network-wide that need to be closely monitored and also eases the bottleneck troubleshooting time and efforts of network operators.

We validated our NTA-CAD scheme using traceroute information and one-way delay measurements collected over 3 months involving 216 paths between the 17 DOE national lab network locations, published via perfSONAR web services. We showed how the APD can be used to generate uncorrelated anomaly events with high accuracy and much less noise, than using traditional SPD, which can be extremely dense and noisy in terms of false alarms, even over short periods of data analysis. Using the critical hop/link based spatial filtering, we were able to conclude that some factor closer to the FNAL edges may be causing several of the correlated anomaly events. Further, we showed that the burstiness information has to be supplemented with the number of common events information in prioritization of critical paths for troubleshooting, and the paths with relatively higher burstiness and common events need to have higher priority during troubleshooting.

As part of our future work, we are interested in using the guidance provided by our NTA-CAD scheme and coupling it with additional information sources (e.g., router logs, maintenance activity logs) within frameworks such as NICE [10] to more effectively determine the ground truth. In addition, we plan to conduct more extensive spatial and temporal filter analysis for other perfSONAR data sets belonging to academia and industry communities as well as other metrics. This will allow us to better understand how network operators can effectively and easily handle several network-wide anomaly event occurences with varying degrees of severity showing up as 'red' alerts on their monitoring dashboards.

References

1. Hanemann, A., Boote, J., Boyd, E., Durand, J., Kudarimoti, L., Lapacz, R., Swany, M., Trocha, S., Zurawski, J.: perfSONAR: a service oriented architecture for multi-domain network monitoring. In: Proceedings of Service Oriented Computing, LNCS 3826, pp. 241–254. Springer (http://www.perfSONAR.net) (2005)
2. Zurawski, J., Swany, M., Gunter, D.: Scalable framework for representation and exchange of network measurements. In: Proceedings of IEEE TRIDENTCOM (2006)
3. Guok, C., Robertson, D., Thompson, M., Lee, J., Tierney, B., Johnston, W.: Intra and interdomain circuit provisioning using the OSCARS reservation system. In: Proceedings of IEEE/ICST Conference on Broadband Communications, Networks, and Systems (2006)
4. Allen, J.: Driving by the rear-view mirror: managing a network with cricket. In: Proceedings of USENIX Network Administration Conference (1999)
5. Matthews, W., Cottrell, L.: The PingER project: active internet performance monitoring for the HENP community. IEEE Commun. Mag. Netw. Traffic Meas. Exp. **38**(5), 130–136 (2000)
6. Zurawski, J., Boote, J. et al.: Hierarchically federated registration and lookup within the perfSONAR framework. In: Proceedings of IFIP/IEEE Integrated Management Symposium (2007)
7. McGregor, A., Braoun, H-W.: Automated event detection for active measurement systems. In: Proceedings of Passive and Active Measurement Workshop (2001)
8. Logg, C., Cottrell, L.: Experiences in traceroute and available bandwidth change analysis. In: Proceedings of ACM SIGCOMM Network Troubleshooting Workshop (2004)
9. Calyam, P., Pu, J., Mandrawa, W., Krishnamurthy, A.: OnTimeDetect: dynamic network anomaly notification in perfSONAR deployments. In: Proceedings of IEEE/ACM MASCOTS (2010)
10. Mahimkar, A., Yates, J., Zhang, Y., Shaikh, A., Wang, J., Ee, C.: Troubleshooting chronic conditions in large IP networks. ACM SIGCOMM CoNEXT (2008)
11. Plonka, D., Barford, P.: Network anomaly confirmation, diagnosis and remediation. In: Proceedings of IEEE Allerton Conference on Communication, Control, and Computing (2009)
12. Palla, G., lszl Barabsi, A., Vicsek, T., Hungary, B.: Quantifying social group evolution. In: Proceedings of Nature (2007)
13. Collaboration, A., Laurens, P., Severini, H., Wolff, S., Lake, A., Kee, S., Zurawski, J., Wlodek, T.: Monitoring the US ATLAS network infrastructure with perfSONAR-ps. In: Proceedings of Conferences on Computing in High Energy and Nuclear Physics (CHEP) (2012)
14. Lakhina, A., Crovella, M., Diot, C.: Diagnosing network-wide traffic anomalies. In: Proceedings of ACM SIGCOMM (2004)
15. Soule, A., Salamatian, K., Taft, N.: Combining filtering and statistical methods for anomaly detection. In: Proceedings of Conference on Internet Measurement (2005)
16. Zonglin, L., Guangmin, H., Xingmiao, Y., Dan, Y.: Detecting distributed network traffic anomaly with network-wide correlation analysis. In: Proceedings of EURASIP J. Adv. Signal Process (2009)
17. Zhou, Y., Hu, G.: Network-wide anomaly detection based on router connection relationships. IEICE Trans. **94-B**(8), 2239–2242 (2011)
18. Yalagandula, P., Lee, S., Sharma, P., Banerjee, S.: Correlations in end-to-end network metrics: impact on large scale network monitoring. In: Proceedings of IEEE INFOCOM Workshops (2008)
19. Yalagandula, P., Lee, S., Sharma, P., Banerjee, S.: Leveraging correlations between capacity and available bandwidth to scale network monitoring. In: Proceedings of IEEE GLOBECOM (2010)
20. Mutt, E., Sharma, M., Soman, J., Kothapalli, K., Mitra, A.: Graph theoretic approach for studying correlated motions in biomolecules. In: Proceedings of IEEE NaBIC (2009)

21. Fernandes, G., Owezarski, P.: Automated classification of network traffic anomalies. In: Proceedings of SecureComm (2009)
22. Wang, T., Srivatsa, M., Agrawal, D., Liu, L.: Spatio-temporal patterns in network events. In: Proceedings of ACM Co-NEXT (2010)
23. Hanemann, A., Jeliazkov, V., Kvittem, O., Marta, L., Metzger, J., Velimirovic, I.: Complementary visualization of perfSONAR network performance measurements. In: Proceedings of IEEE International Conference on Internet Surveillance and Protection (2006)
24. Bruackhoff, D., Dimitropoulos, X., Wagner, A., Salamatian, K.: Anomaly extraction in backbone networks using associated rules. IEEE/ACM Trans. Netw. **20**(6), 1788–1799 (2012)
25. Chan, E., Luo, X., Fok, W., Li, W., Chang, R.: Non-cooperative diagnosis of submarine cable faults. In: Proceedings of Passive and Active Measurement Conference (PAM) (2011)
26. Dainotti, A., Pescape, A., Ventre, G.: Worm traffic analysis and characterization. In: Proceedings of IEEE International Conference on Communications (ICC) (2007)
27. Dainotti, A., King, A., Claffy, K., Papale, F., Pescape, A.: Analysis of a "/0" stealth scan from a Botnet. In: Proceedings of ACM SIGCOMM/SIGMETRICS Internet Measurement Conference (IMC) (2012)
28. Calyam, P., Krymskiy, D., Sridharan, M., Schopis, P.: Active and passive measurements on campus, regional and national network backbone paths. In: Proceedings of IEEE ICCCN (2005)
29. Grigoriev, M., Demar, P., Eads, D., Tierney, B., Metzger, J., Lake, A., Frey, M., Calyam, P.: E-Center: Collaborative platform for the wide area network users. In: Proceedings of Conferences on Computing in High Energy and Nuclear Physics (CHEP) (2012)
30. Dhanapalan, M., Calyam, P., Sridharan, M.: Nagios adaptive plateau anomaly detection (APD) Plugin-v1.4. http://anonsvn.internet2.edu/svn/perfSONAR-PS/branches/osc-APD-Nagios/perfSONAR_PS-Nagios/doc/APD_README.txt (2012)
31. Marchetta, P., Merindol, P., Donnet, B., Pescape, A., Pansiot, J.-J.: Topology discovery at the router level: a new hybrid tool targeting ISP networks. IEEE J Sel. Areas Commun. (JSAC) **29**(9), 1776–1787 (2011)
32. Donato, W., Marchetta, P., Pescape, A.: Detecting third-party addresses in traceroute IP paths. In: Proceedings of ACM SIGCOMM (2012)
33. Ellson, J., Gansner, E., Koutsofios, E., North, S., Woodhull, G.: Graphviz and dynagraph—static and dynamic graph drawing tools. In: Proceedings of Graph Drawing Software, pp. 127–148. Springer (2003)

Author Biographies

Prasad Calyam received his B.E. degree in Electrical and Electronics Engineering from Bangalore University, India and M.S. and Ph.D. degrees from the Department of Electrical and Computer Engineering at The Ohio State University in 1999, 2002 and 2007, respectively. He is currently an Assistant Professor in the Department of Computer Science at University of Missouri-Columbia. His research and development areas of interest include: Distributed and Cloud Computing, Computer Networking, Networked-Multimedia Applications, and Cyber Security.

Manojprasadh Dhanapalan received the M.S. and B.E. degrees in Computer Science and Engineering from The Ohio State University and Sri Venkateswara College of Engineering in 2012 and 2009, respectively. He contributed to this paper during his tenure as a Graduate Research Associate at the Ohio Supercomputer Computer. His current research interests include distributed systems and performance measurement.

Mukundan Sridharan received the B.S. degree in Electronics and Communication Engineering from Madras University, India, and the M.S. and Ph.D. degrees in Computer Science and Engineering from The Ohio State University, in 2000, 2002, and 2011, respectively. He is currently a Ubiquitous Computing Engineer at The Samraksh Company. His current research interests include wireless sensor networks, multimedia networking, and distributed systems.

Ashok Krishnamurthy earned his bachelors degree in Electrical Engineering in 1979 from the Indian Institute of Technology in Madras, India. He received his Masters and Doctorate degrees in Electrical

Engineering at the University of Florida in 1981 and 1983, respectively. He is currently a Deputy Director of Renaissance Computing Institute (RENCI). His research interests include signal/image processing, high performance computing, parallel high-level language applications and computational models of hearing.

Rajiv Ramnath received his Doctorate and Masters degrees in Computer Science from The Ohio State University and his Bachelors degree in Electrical Engineering from the Indian Institute of Technology. He is currently Director of Collaborative for Enterprise Transformation and Innovation (CETI) at The Ohio State University. His research interests include wireless sensor networking and pervasive computing to business-IT alignment, enterprise architecture, software engineering, e-Government, collaborative environments and work-management systems.