Fine-Grain Diagnosis of Overlay Performance Anomalies Using End-Point Network Experiences

Fida Gillani*, Ehab Al-Shaer*, Mostafa Ammar† and Mehmet Demirci †
*University of North Carolina Charlotte (UNCC)
Charlotte, North Carolina, USA
{sgillan4, ealshaer}@uncc.edu
† School of Computer Science,
Georgia Institute of Technology, Atlanta, Georgia, USA
{ammar, mehmet}@cc.gatech.edu

***Abstract*—Overlay networks were proposed to improve Internet reliability and facilitate a rapid deployment of new services. Non-invasive diagnosis of performance problems is the key capability for overlay service management in order to adapt to dynamic network conditions in a timely manner. Existing overlay diagnosis approaches assume extensive knowledge about the network, and require monitoring sensors or active measurements. In this paper, we propose a novel diagnosis technique to localize performance anomalies and determine the packet loss contribution for each network component. Our approach is purely based on endpoint packet loss observations to reason about the location of observed packet loss without active probing or sensor deployment. We formulate the problem as a constraint-satisfaction problem using constraints derived from network loss invariants and end-user observations. Our solution also circumvents the possibilities of insufficient or malicious end-user participation. We evaluate our approach extensively using simulation and experimentation, and demonstrate the accuracy, effectiveness and scalability of our approach for various network sizes, participation levels and spurious amounts.**

I. Introduction

As the decentralized structure of the Internet increases its scalability, it makes Internet problems harder to analyze and diagnose. Overlay networks were proposed to improve the reliability [1] of the Internet and to facilitate performance sensitive services, such as mission critical networks, high performance cloud computing etc., that are difficult or impossible to deploy natively on the Internet [2], [3]. As overlays extend the native infrastructure with new functionality, potential of overlay problems also increases beyond traditional networks for many reasons. First, a native network problem will eventually propagate to overlays unless problem determination and dynamic reconfiguration are performed. Second, overlay nodes usually incorporate much more complex functionality than basic store-and-forward, which makes them more susceptible to hardware/software bugs, component failures, human operational errors, or malicious attacks [4]. Third, most of public overlay networks may be built out of loosely coupled, less trustworthy and less performance capable end-hosts as compared to Internet routers [4]. Fourth, overlay problems are usually manifested as multi-layer problems that might be more complex to analyze and isolate. Efficient service management of overlay network depends upon its capability to non-invasively diagnose performance problems to adapt to dynamic network conditions in a timely manner.

Many solutions have been proposed to diagnose performance problems (called performance anomalies). However, they either require massive deployment of network sensors which may sometimes be infeasible due to the requirement of placing sensors in the core of the network [3]–[9], or they might burden the underlying network infrastructure with additional synthetic traffic for active probing [10]–[13]. There exist some hybrid solutions but they mainly assume extensive network knowledge like posterior or prior fault probabilities of underlying networks [14]–[19] that limit their effectiveness in real practice.

In our approach, end-points (end-users) only need to share their anomalous performance experience (called negative symptoms) by reporting their observed packet loss and traceroute information. Our main contribution in this paper is creating a constraint-based model finder for localizing packet loss in large-scale networks accurately and without requiring active probing or network-based sensors. This model exploits the correlation in the end-user negative symptoms, due to shared symptoms and path segments, in conjunction with identified network loss invariants to encode constraints using Satisfiability Modulo Theories (SMT). Advanced SMT solvers such as Z3 [20] and Yices [21] can solve tens of thousands of constraints and millions of variables [22]. This makes the approach scalable to a large number of evidences (symptoms) and network components. Our model finder identifies the minimum set(s) of root cause network components (e.g., routers) that satisfy the end-user reported observations and network loss invariants. Such a root cause diagnosis based on end-user observations inherits an inevitable risk of low accuracy due to the potential insufficient and/or malicious/spurious end-user observations. Our model circumvents these issues by considering the prominence of contribution of each component in satisfying the entire set of end-user observations. We evaluate the performance of our model using simulation and experimentation, and demonstrate the accuracy, effectiveness and scalability of our approach for various network sizes,

978-3-901882-48-7

participation levels and spurious amounts.

In this paper, we have two main contributions: 1) Our proposed solution only requires end-user network performance observations to localize performance anomalies and to determine the packet loss contribution of each network component; 2) We formulate this as constraints satisfaction problem with method compensating insufficient and/or spurious end-user observations.

The rest of this paper is organized as follows, we explain the difference of our solution with existing techniques in Section II followed by a detailed description of our model with network loss invariants in Section III. We evaluate our proposed solution through simulation and experimentation in Section IV, which is followed by the conclusion in Section V.

II. Related Work

Fault/Anomaly diagnosis and localization have been given significant attention by the research community and we can divide these into three categories: (1) Probing based approaches, (2) Sensors based approaches, and (3) Hybrid approaches.

A. Probing Based Approaches

Solution presented in [11] uses posterior probabilities to identify path(s) to probe at any given time and compares the observed performance over these path(s) with performance thresholds defined in SLAs to raise alarms. Once, an anomaly is detected, it iteratively identifies additional probing paths to localize the effect of the anomaly. This approach is too network oriented, this makes it susceptible to miss intermittent anomalies in highly dynamic overlay networks. Multicast and unicast based tomography technique [12] sends probes to different nodes in a network, acquire probe success information from these nodes, and use complete topology information to trace back footsteps of these probes. The accuracy of this approach is limited to the amount of probe information acquired from nodes, especially in highly dynamic overlays. Another approach [10] calculates the packet loss of all network paths by only monitoring a subset of all network paths. This approach requires full participation of all overlay nodes to provide an initial network topology. Furthermore, it is sensitive to intermittent overlay faults and high churn rate (i.e., overlay nodes join/leave frequently). Moreover, the monitoring of overlay nodes in [10] can be very intrusive even with low probing traffic frequency and may not be suitable for large service providers. And even with such intrusive probing, intermittent overlay network faults cannot be caught.

B. Sensors Based Approaches

A sensor based solution in [7] constructs symptom-fault causality graph using underlay and overlay statistics. Scalability of such an approach is limited in complex, multi-layer and dynamic overlays. The technique in [5] attempts to achieve link granularity diagnosis by passively monitoring network paths that contain every pair of links at least once. However, even when using the proposed optimization approach [5], the number of monitoring probes may be still significantly affected by network topology (e.g., a tree network topology has many branches, and each branch has many links) and could be still large. Another technique [23] uses bloom filter to compare all packets received by different recipients from same multicast application. Bloom filters are intrusive in nature and synchronization of these bloom filters among different recipients is a known issue especially, if the overlay network is highly dynamic. The event-based fault diagnosis approach [7] assumes availability of the symptom-fault casuality relationship likelihood and network-level alarms from the network components, which are rarely available in dynamic overlay networks. Moreover, it is network-centric rather than user-centric, because it allows for diagnosing shared problems such as shared congested links in the networks but not necessarily determine the root cause of problems faced by a single user such as an end-server.

C. Hybrid Approaches

The solution in [18] analyzes end-user observations for fault diagnosis, but the definition of a fault is limited to binary conditions (*e.g.*, unreachable or reachable). It does not investigate the performance loss distribution in the network, but the discrete faults. Bayesian Belief Networks (BBNs) are widely adopted in hybrid approaches [6], [16], [19] to construct symptom-causality graph; however, such approaches require posterior probabilities and knowledge of complete topology to understand the dependency between faulty network components (e.g., routers) and observable symptoms (e.g., network reachability). In [4], the Markov Chain Monte Carlo algorithm is used to identify lossy links in a network with tree topology based on passive traffic observations at a server.

D. Our Approach

Our solution proposes an approach by reasoning about network problems assuming little or no knowledge about the network, and without requiring any monitoring sensors or active measurements. Leveraging the power of information sharing, we investigate a purely passive approach that relies only on end-user observations (bad symptoms) to diagnose network performance problems and determine the contribution of each network component, in an accurate and scalable manner.

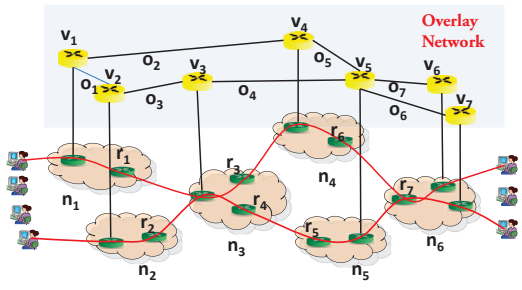


Fig. 1. Overlay Network Model.

TABLE I
VARIABLES USED AND THEIR DEFINITIONS

Notation	Definition
c_j	j^{th} Overlay component
C	Set of all components c_j
e_i	i^{th} Negative performance evidence
E	Set of all evidences e_i
p_i	i^{th} Overlay path
θ_i	Total packets reported as sent in e_i
κ_i	Packets successfully delivered by e_i
d_i	Percentage of packet loss of evidence e_i
E_{c_j}	Set of evidences that contains c_j
D_{c_j}	Set of all packet losses reported for c_j
f_i	i^{th} Anomaly scenario
F	Set of all anomaly scenarios
$\chi(\cdot)$	Boolean function
δ_j	Packet loss contribution of component c_j
t	Fixed time interval to report evidences
τ	Component tuple inside an anomaly scenario

III. Constraint Satisfaction Problem using Satisfiability Modulo Theory

A. System Overview

We model an overlay network as a directed graph $G =< V, O >$, comprising a set of overlay nodes V as vertices joined by a set of bidirectional overlay links O, as shown in Figure 1. An *overlay link* connects two overlay nodes v_i and v_j ($v_i, v_j \in V$). An *overlay path* p_i may consist of multiple overlay links, which traverses a sequence of overlay components. Here, an *overlay component* could be a router (r) or an overlay node (v). For example, in Figure 1, the components contained by the overlay path p_1 between $n1$ and $n6$ can be represented in a router-level, as, $p_1 = \{v_1, r_1, v_3, r_3, v_4, r_6, r_7, v_6\}$. Every packet in overlay network traverses one of these overlay paths and either it gets through successfully or dropped by any overlay component along the path. We call this the overlay experience of an end-user and it is either of the following two types:

Definition 1: An end-user observed negative experience is called a *negative performance evidence* e_i, which represents total number of packets θ_i, sent over an overlay path p_i and total packets successfully delivered κ_i in a time interval t. Thus, $e_i = \{c_1, c_2, \cdots, c_n\}$, where c_j is an overlay component. The percentage of packets dropped d_i reported in e_i is then $d_i = \frac{\theta_i - \kappa_i}{\theta_i}$. These negative performance evidences are collected at the end of fixed interval (globally set) t, which is typically set to 30 seconds in our experiments. We consider that, this interval is small enough compared to the duration for which a performance anomaly remain persistent in the network, which typically lasts for a long time, even sometimes, for an hour [10], [24]. Therefore, synchronization of these end-user reporting intervals is not needed.

Definition 2: An end-user observed positive experience is called a *positive performance evidence* e_i, which represents a successful session in time interval t with no packet loss, *i.e.*, $d_i = 0$. Thus, $e_i = \{c_1, c_2, \cdots, c_n\}$, where c_j is an overlay component.

The end-users will only need to share their negative performance evidences in order to identify the bottleneck component(s) in the overlay network. We denote E as the collection of all negative performance evidences, $e_i \in E$ and C as the set of all components in the network, $c_j \in C$. We denote a relevant performance evidence set $E_{c_j} \subseteq E$ of a component c_j as the set of different end-user negative performance evidences that contains the component c_j, i.e., $\forall e_i \in E_{c_j},\ c_j \in e_i$. We consider all $c_j \in e_i$ equally likely to be responsible for the packet loss d_i. A relevant packet loss set D_{c_j} of a component c_j is the set of different percentages of packet losses reported by different evidences for component c_j, i.e., $\forall e_i \in E_{c_j},\ d_i \in D_{c_j}$. We denote the component visibility as the number of end-user evidences in E_{c_j} *i.e.*, $|E_{c_j}|$, for example, if a component c_j appears in evidences from $e_1, \ldots, e_{10}$, then the visibility of component c_j is $|E_{c_j}| = 10$. We denote the percentage of packets dropped by component c_j in time interval t as δ_j, by dividing total number of packets dropped by component c_j in t with total packets received by the component c_j in t, as $\delta_j = \frac{\text{packets dropped}}{\text{packets received}}$, detail discussion is in Section III-C.

B. Formulation of Satisfiability Problem

We employ the concept used in boolean tomography methods [12] to represent the status of a component. For each negative performance evidence e_i, we assign a *bad* label to the corresponding overlay path p_i; for each *bad* overlay path there is at-least one *bad* (anomalous) component along the path dropping packets, $c_j \in e_i$. Otherwise, A component is labeled as *good*.

Definition 3: The working status of a component c_j is called the condition of the component c_j. Component c_j is called in bad (anomalous) condition if it is responsible for packet loss reported by the evidence e_i, such that, $c_j \in e_i$; otherwise, component c_j is called in good condition.

Fine grain diagnosis of performance anomalies using end-user negative performance evidences is essentially to evaluate the condition of each potential anomalous component and to find a set of bad components that can collectively explain packet loss distribution of all evidences.

Definition 4: Let E be a set of end-user negative performance observations containing all components labeled *bad*. An anomaly scenario, f, is a set of anomalous components such that they cover all evidences and collectively explain the packet loss distribution of all evidences. An anomaly scenario can be formally defined as:

$$\forall e_i \in E, \exists\ c_j \in f, c_j \in e_i, Explain(c_j, d_i) \Leftrightarrow FS(f, E) \tag{1}$$

where *Explain* and *FS* are predicates, $Explain(c_j, d_i)$ means the component c_j partially or completely explains the packet loss d_i reported by evidence e_i and $FS(f, E)$ means f is an anomaly scenario of E.

Evidential Reasoning Invariant: An anomaly scenario basically renders a constrained logical relationship between E and all its components. The *logical relationship* aspect

of anomaly scenario implies there exists at-least one *bad* component in each evidence and *constrained* implies we are only interested in such a logical relationship that also explains packet loss distribution of all evidences. This logical relationship between an evidence and its related components can be represented as a boolean function: $\chi(e_i) = \bigvee_{c_j \in e_i} \chi(c_j)$ which means the evidence e_i will be true if any of the related component $c_j \in e_i$ is bad. Thus, we can also represent E as a boolean function as follows:

$$\chi(E) = \bigwedge_{e_i \in E} \; (\bigvee_{c_j \in e_i} \chi(c_j)) \tag{2}$$

This means E will only be satisfied (true) if all evidences $e_i \in E$ are satisfied (true), which is essentially an evidential reasoning invariant. The problem of finding a set of bad components that can satisfy all evidences, as in Equation 2, is an NP-complete problem [18] and can be mapped to SAT [25] or hitting set problem [2]. This evidential reasoning invariant in Equation 2 only establishes a logical relationship between E and its components and does not necessarily comply to network loss invariants *e.g.*, total packet loss contribution of all bad components in an evidence e_i, as identified by evidential reasoning invariant, may not necessarily satisfy the packet loss d_i reported by the evidence e_i. These network loss invariants are discussed as follows.

C. Network Loss Invariants

Packet drop is normally the product of congestion at components (routers) in the network. We assume, to respond congestion, components along a path use fair drop policy to drop packets from each flow, which is proportional to the size of the flow. Fair drop is optimum for high-speed transit networks and it is also considered a priori in other approaches [26], [27]. Fair drop capacitates networks to exhibit certain network loss invariants, which we use as constraints in our SMT based model to limit the search space of possible logical relationships that exists between E and its components.

1) Anomalous Component Validity Constraint: If a component c_j in an evidence e_i is considered bad, then the packet loss contribution of that component, δ_j should be positive, as a component with no packet loss is not *bad*.

$$\chi(c_j) \rightarrow \delta_j > 0 \tag{3}$$

2) Shared Path Constraint: Two performance evidences e_i and e_k are correlated if $e_i \cap e_k \neq \emptyset$ and this correlation also extends to the packet loss experienced by these evidences. This constraint states: *If two performance evidences, e_i, e_k, share the same overlay path $p_i \Leftrightarrow p_k$ then they experience same percentage of packet loss, $d_i = d_k$*, formally defined as:

$$(e_i \Leftrightarrow e_k \rightarrow d_i = d_k) \wedge (e_i \subset e_k \rightarrow d_i \leq d_k) \tag{4}$$

For example, evidences e_i and e_k have only one component c_j, *i.e.*, $e_i = e_j = \{c_j\}$ and if the component c_j contributes packet loss δ_j then it is the same for both evidences, $\delta_j = d_k = d_i$. This property reveals another important fact that if $e_i \subset e_k$ then at-least one component in e_k but not in e_i, *i.e.*, $c_j \in (e_k - (e_k \cap e_i))$ is responsible for the packet loss $d_k - d_i$.

3) Maximum Loss Contribution of Component Constraint: The relevant packet loss set of component c_j is D_{c_j} and we can designate a certain loss contribution $d_i \in D_{c_j}$ as the maximum possible loss contribution of a component, such that, *A component c_j cannot be accused of more than the minimum of packet loss contributions reported for the component c_j*. Formally it is defined as:

$$0 \leq \delta_j \leq \min(D_{c_j}) \tag{5}$$

For example, if evidence e_i has only one component c_j, then loss reported by the evidence is equal to the loss contribution of the component, $d_i = \delta_j$. If there exists an evidence $e_k \in E_{c_j}$, where $E_{c_j} = \{e_i, e_k\}$, reporting higher packet loss than evidence e_i, then there must be at-least one component other than the component c_j in evidence e_k that is also dropping packets. The relevant packet loss set of component c_j is $D_{c_j} = \{d_i, d_k\}$ and the maximum loss contributed by the component c_j can not be more than d_i which is the minimum.

4) Aggregated Loss Contribution Constraint: All components in an evidence in bad condition should collectively explain the reported packet loss d_i of evidence e_i, but the aggregate of loss contributions of all components is given as: *The aggregate packet loss contribution of all components in an evidence $c_j \in e_i$ is greater than or equal to the packet loss percentage d_i reported by the evidence e_i*, formally defined as:

$$\sum_{c_j \in e_i} (\chi(c_j) * \delta_j) \geq d_i \tag{6}$$

For example, packets are dropped sequentially from source to destination over an overlay path. If two components $c_j, c_l \in e_i$ along a path p_i drop packets with same percentage $\delta_j = \delta_l = 10\%$, and if total packets sent are, $\theta_i = 100$, then $d_i = 19\%$ which is less than aggregate loss contributions of both components c_j and c_l, *i.e.*, $\delta_j + \delta_l = 20\%$.

5) Packet Loss Conservation Constraint: The aggregate successfully delivered traffic, calculated based upon loss contribution for each component, satisfies the reported successful traffic for each evidence in the system, formally defined as:

$$\theta_i * \prod_{c_j \in e_i} (1 - (\delta_j * \chi(c_j))) = \kappa_i \tag{7}$$

6) Set of Anomalous Component Size Constraint: Normally, during any given time interval the fraction of network dropping packets is small [6], [10], which implies, an anomaly scenario with minimum number of components should be preferred. Let, $\mathcal{F}$ be the set of all satisfiable anomaly scenarios, then formally this property is defined as:

$$\forall \; f \in \mathcal{F}, \; |f| \leq Threshold \tag{8}$$

We use *Threshold* instead of minimum size to introduce this as a tunable parameter to expand the search space for

potential anomaly scenarios. But we have used minimum size as *Threshold* for our evaluation in Section IV.

D. Constraints Satisfaction Problem using SMT

We formulate the network loss invariants discussed in Section III-C in conjunction with evidential reasoning invariant in Equation 2 as constraints using satisfiability modulo theories. The final constraints satisfaction problem is given as follows:

$$\chi(E) = \bigwedge_{e_i \in E} \; (\bigvee_{c_j \in e_i} \chi(c_j)) \tag{9}$$

$$\bigwedge_{e_i \in E} \; (\theta_i * \prod_{c_j \in e_i} (1 - (\delta_j * \chi(c_j))) = \kappa_i) \tag{10}$$

$$\forall \; c_j \in E \;\; 0 \leq \delta_j \leq \min(D_{c_j}) \tag{11}$$

$$\forall \; e_i \in E \;\; \sum_{c_j \in e_i} (\chi(c_j) * \delta_j) \geq d_i \tag{12}$$

$$\forall \; f \in F, |f| \leq Threshold \tag{13}$$

$$\forall \; e_i, e_k \in E, e_i \Leftrightarrow e_k \rightarrow d_i = d_k \wedge \tag{14}$$

$$e_i \subset e_k \rightarrow d_i \leq d_k$$

$$\forall \; c_j \in C, \chi(c_j) \rightarrow \delta_j > 0 \tag{15}$$

The SMT based model generates a single satisfiable solution, whereas, we are interested in finding all satisfiable solutions. Every time the model output a satisfiable solution, we take the complement of the solution and assert it back to the model. This forces the model to generate a new satisfiable solution after every iteration. We keep repeating this process until we get all satisfiable solutions (anomaly scenarios).

E. Plausible Reasoning for Multiple Anomaly Scenarios

Our solution may potentially produce multiple candidates of anomaly scenarios. There are number of conditions that contribute to increasing the size of the anomaly scenario candidates. First, *insufficient evidences* due to low participation can potentially increase the number of scenarios. This can be attributed to following factors: (1) the number of participating users, (2) the distribution/correlation of evidences with respect to network paths, and (3) the potential of evidence loss. Both the number of participating users and potential correlation between the evidences, due to the path overlapping, significantly impact the likelihood that an anomalous network component can be observable by multiple users. These factors will be extensively evaluated in Section IV. Second, *spurious evidences*, which are either due to malicious or over-sensitive end-users, can also increase the possibilities of satisfiable anomaly scenarios. Due to such uncertainty, there is an inevitable risk in the process of decision making when there are multiple satisfying anomaly scenarios. Thus, we have developed a heuristic based approach to select an anomaly scenario with the highest total belief based on the components' *visibility* and *frequency* [28]. The component visibility in this case reflects the contribution of this component on the satisfiability of the entire evidence set, E. However, the frequency factor reflects the prominence of the loss contribution of this component in all anomaly scenarios, F. We define a new parameter β called *total belief* of an anomaly scenario. We calculate the belief of an anomaly scenario by aggregating the likelihood of all items in an anomaly scenario. We call the item in an anomaly scenario as *component tuple* (τ) that comprises of a component c_j and its loss contribution δ_j as, $< c_j, \delta_j >$.

In addition, this combination of both visibility and frequency in calculating the belief metric of each component tuple, will help in damping the affect of *spurious observations* due to malicious or over-sensitive reports. Detailed evaluation is provided in Section IV-C2. In our heuristic approach we, first, determine a set of distinct tuples[1], S, as $S = \bigcup_{f_m \in F} f_m$. Then the belief of each tuple β_τ is calculated by multiplying its frequency with its component visibility, $\forall \tau_x \in S, \beta_{\tau_x} = |F|_{\tau_x} * |E_{c_j}|$, where, $c_j \in \tau_x$. Finally, we select the anomaly scenario with maximum total belief of its tuples as the most probable candidate anomaly scenario C_d. This is formally showed in the following equation:

$$C_d = \max[\forall f_m \in F \sum_{\tau_x \in f} \beta_{\tau_x}] \tag{16}$$

It is very unlikely to have multiple anomaly scenarios with the same total belief value. However, if this is the case, we include in the final solution the common components associated with loss contribution ranges based on the contributions in the original anomaly scenarios with same maximum total belief. Let M be a set of all anomaly scenarios of the same maximum total belief, then the variant of Equation 16 is formally defined as follows:

$$C_d = \bigcup_{c_j \in M} < c_j, \bigcup_{\delta_j \in M} \delta_j > \tag{17}$$

IV. Implementation of Simulation and Evaluation

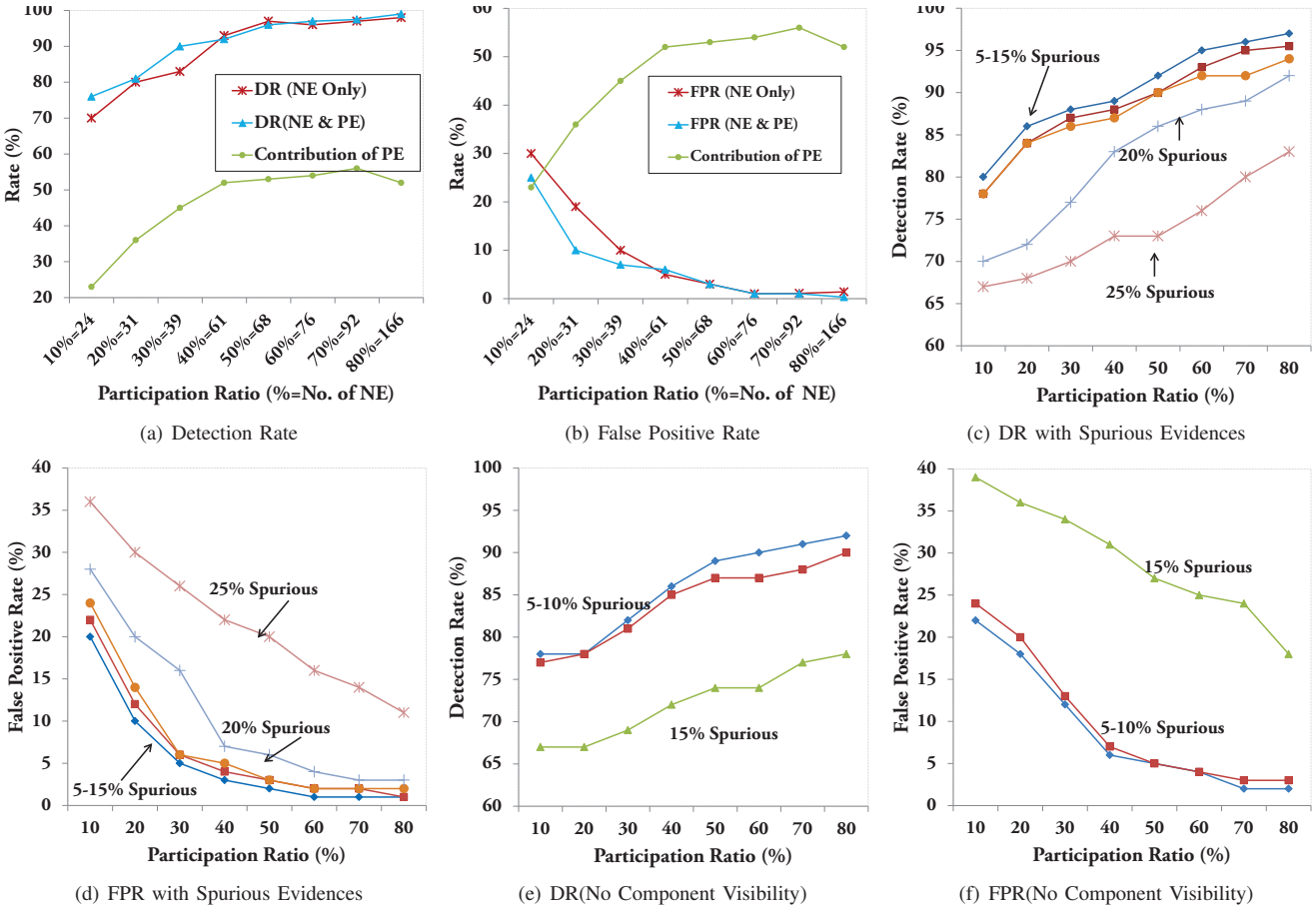
In this section we discuss evaluation metrics, simulation methodology and evaluation of our technique.

A. Evaluation Metrics

The performance evaluation metrics include: 1) Detection Rate (DR); and 2) False Positive Rate (FPR). Let, C_f be the actual anomaly scenario and C_d be the detected anomaly scenario. Detection rate will be $DR = \frac{|C_d \cap C_f|}{|C_f|}$, where $|C_d \cap C_f|$ gives the number of common component tuples (successfully detected) in both actual and detected anomaly scenarios. Similarly, False Positive rate is $FPR = \frac{|C_d| - |C_d \cap C_f|}{|C_d|}$, which gives the fraction of the detected anomaly scenario wrongly accused of being anomalous.

Every time we inspect performance we also calculate network correlation to establish their relationship. We calculate correlation of an evidence e_i by aggregating its degree of overlapping with other evidences and normalizing it, $Corr(e_i) = \frac{\sum_{e_k \in E} |e_i \cap e_k|}{|E|}$, where $i \neq k$.

[1]As loss contribution can be a real value, we use $\delta_j \pm \Delta$ when compares different loss contribution, where Δ is very small.



(a) Detection Rate (b) False Positive Rate (c) DR with Spurious Evidences

(d) FPR with Spurious Evidences (e) DR(No Component Visibility) (f) FPR(No Component Visibility)

Fig. 2. Figures 2(a) and 2(b) show effect of Participation Ratio in Topology with 100 nodes and 10 performance anomalies, *PE* is Positive Evidence, *NE* is Negative Evidence, and *Contribution of PE* represents fraction of components labeled non-anomalous by positive evidences. *X-axis* in Figures 2(a) and 2(b) shows Participation Ratio in % and the number of NE it represents, respectively.Figures 2(c), 2(d), 2(e) and 2(f) show the effect of changing spurious evidences on performance with and without using Component visibility to calculate total belief of each evidence, only considering NE.

B. Simulation Implementation and Methodology

Our system simulation consists of three modules: 1) Topology generation and overlay mapping; 2) Evidence generation; and 3) Constraints satisfaction problem generation. In topology generation module we use BRITE [29] to generate router level topologies, which uses two types of models to interconnect nodes, preferential and random. BRITE uses two types of models for network growth and node interconnection, (1) Waxman model, that generates a random topology using Waxmans probability to interconnect network nodes, and (2) BarbarasiAlbert model proposed by Barabasi and Albert [29], it generates power law out-degree distribution using incremental network growth and preferential node connectivity as network growth models. We use random topology generator using Waxman probability model because in Barabasi model based network network paths are more correlated, which adds a bias in the network. In overlay mapping, we start by randomly designating a small fraction of routers with least degree as ingress and egress routers and then randomly distribute end-users to these ingress and egress routers as sources and destinations, respectively. We calculate shortest paths between all sources to all destinations. We randomly select routers with least degree and designate these as overlay nodes. After this we recalculate all paths from all sources to all destinations flowing through these overlay nodes and denote these as *total paths*.

During evidence generation, we select a sub-set of *total paths* and designate these as active overlay paths[2]. We use the similar loss distribution model as in [16] and distribute the failure probabilities among independent components as: 1) uniform fault distribution, which says the performance anomalies are randomly distributed throughout the network. To simulate this behavior we use the probabilities in range [0.001, 0.01] for all components; (2) differential fault distribution, which says performance anomalies occur in clusters and are co-located rather than evenly distributed in the network. To simulate this behavior we use probabilities in range [0.001, 0.01] for substrate (underlay) components (e.g., routers), but [0.01, 0.1] for overlay components. We randomly assign different traffic volumes (number of packets) ranging from $1K$ to $100K$ to end-users and simulate loss process using fair drop

[2]Active overlay paths are paths that are carrying traffic.

mechanism. This generates two set of end-user observations: 1) negative evidences, active paths with packet losses; and 2) positive evidences, active paths with no packet losses. We use these active paths and consider it as participation ratio.

We use $Z3$ theorem prover to formulate a constraints satisfaction model from generated evidences [20]. $Z3$ is a state-of-the art theorem prover from Microsoft Research and it is used to check the satisfiability of logical formulas over one or more theories.

We have used a Core 2 Duo machine with 2.3 GHz processor and 4GB RAM to run all experiments explained in the following section. None of the experiment take more than 1 minute to generate result, therefore, we do not report computation time results.

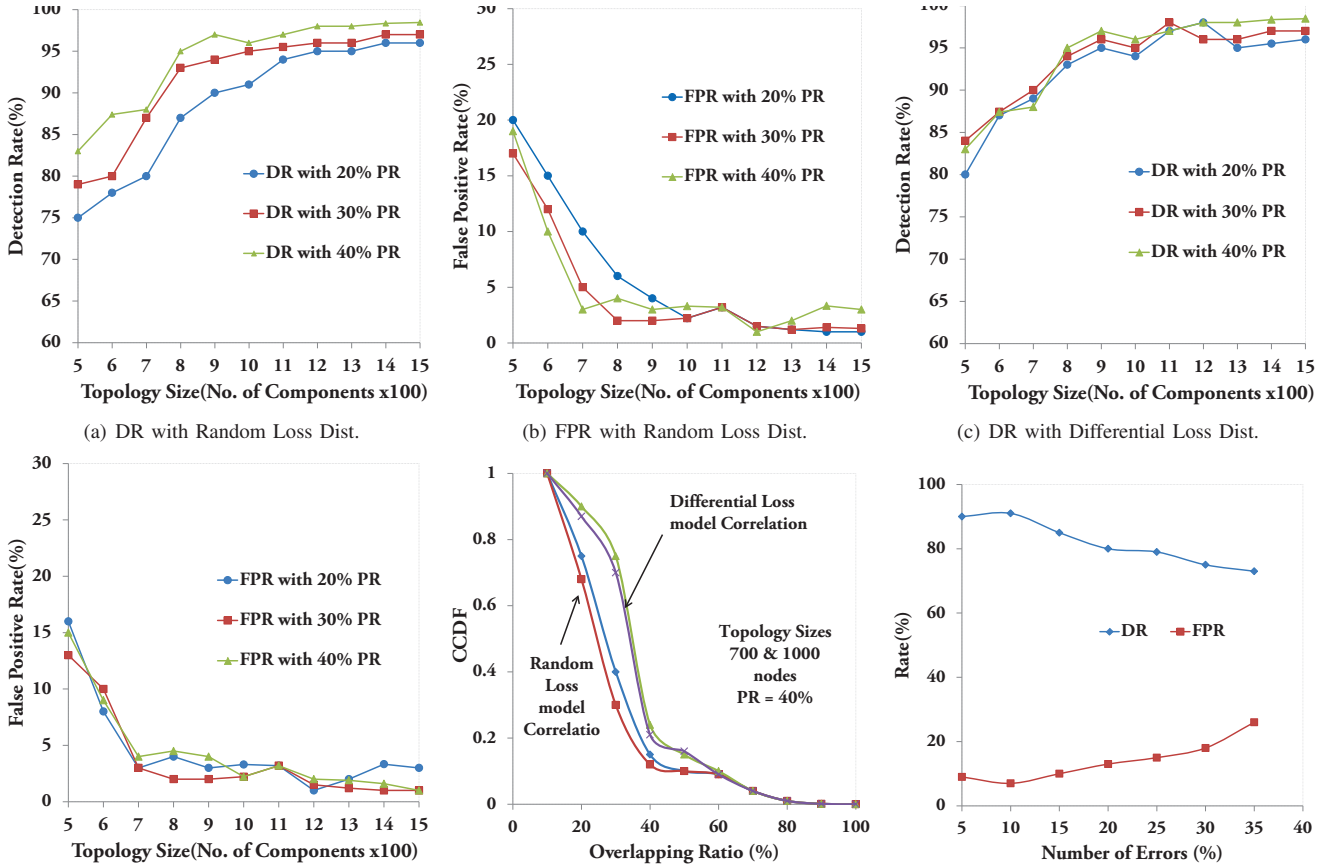
C. Evaluation of Simulation

1) Effect of Participation Ratio on Performance: As mentioned in Section III-E the accuracy of our technique is sensitive towards the participation ratio that determines the number of negative evidences available. To evaluate the sensitivity, we analyze the performance of our technique by changing the participation ratio between 10-80%, and results are showed in Figure 2(a) and Figure 2(b), with a topology of size 100 nodes and 10 performance anomalies. For these experiments we have used small topology to show the impact of insufficient evidences on performance, because in large topologies even a small participation ratio would result into large number of negative evidences. The *X-axis* in Figure 2(a) and Figure 2(b) is showing two types of information. The number with % sign represents the participation ratio used in the corresponding experiment and the number after = represents the equivalent number of negative evidences, *e.g.*, at 10% and 80% participation ratio, the actual received negative evidences were 24 and 166, respectively. We ran these experiments with and without considering positive evidences. As mentioned earlier, positive evidence means there is no packet loss over that path, which means all components along the path are not dropping any packets. Therefore, if a component has positive evidence this means it is non-anomalous. We use positive evidences to analyze the dependency of our technique on such information. Both Figures 2(a) and 2(b) show the fraction of total components flagged non-anomalous due to positive evidences at each step, *e.g.*, at 30% participation ratio, almost 35% overlay components were flagged as non-anomalous. As showed in Figure 2(a), at only 39 negative evidences (30% participation ratio), without positive evidences, detection rate reaches to 80% and in Figure 2(b) false positive rate reaches to 10%. When both positive and negative evidences are considered, performance only improves under low participation even though almost 30-50% components are flagged as non-anomalous, as in Figures 2(a) and 2(b). These results suggest that our technique is not dependent upon the availability of positive evidences, though it helps improving the detection rate and reducing the false positive rate at low participation ratio. Also, our technique provides very acceptable detection rate and false positive rate even on low participation ratio.

2) Effect of Spurious Evidences on Performance: Spurious evidences cause uncertainty in deciding performance anomalies, thus results into more anomaly scenarios. Spurious evidence can be a negative evidence notifying incorrect percentage of packet loss or a positive evidence posing as a negative evidence with percentage of packet loss greater than 0. To evaluate the impact of evidential spuriousness on our approach, we use positive evidences as spurious evidences. We randomly select positive evidences, assign them a fake packet loss percentage from 5% to 10%, and use these as spurious evidences. We have varied their amount by 5-25% of total evidences but keeping the number of performance anomalies fixed at 5% of network size in Figures 2(c) and 2(d). Results in Figure 2(c) show that detection rate is resilient to spurious evidences as long as it is less than or equal to 20% of total evidences. The impact of evidential spuriousness is quite visible under low participation, as can be seen in both Figures 2(c) and 2(d). In Figure 2(d) false positive rate shows similar resilience up-till 20% spurious participation, after that at 25% spurious participation, false positive rate reaches to almost 30%.

In Section III-E we combine the frequency of component tuples in anomaly scenarios with its component visibility to cater for the effect of spurious evidences. In Figures 2(e) and 2(f), we show that if we do not use component visibility with component tuple frequency, both detection rate and false positive rate suffers a lot even after just 10% of spurious participation.

3) Effect of Change in Loss Distribution Model and Topology Size on Performance Scalability: Figure 3 shows the performance scalability with respect to increase in size of a network. To evaluate this we use networks with both types of loss distribution models, random loss distribution and differential loss distribution models, as explained in Section IV-B. We select 5% of total components as performance anomalies (dropping packets) in all topologies of sizes 500-1500 nodes and analyze the performance by varying participation ratio from 20-40%. Because of the large topology sizes, even 20% participation ratio generates sufficient evidences that is why detection rate in Figure 3(a) and Figure 3(c) is sufficiently high, almost 90% and 95%, respectively. Similarly, false positive rate in Figure 3(b) and Figure 3(d) is significantly low, 10% and 5%, respectively, regardless of loss distribution model. In Figure 3(c), the detection rate when anomalies are distributed using differential loss distribution model is 5-10% better than the detection rate when anomalies are distributed randomly in Figure 3(a). Same 5-10% reduction in false positive rate is also observable in Figure 3(d) and Figure 3(b) between differential and random loss distribution models, respectively. It is because anomalies are more co-located when distributed differentially as compared to random distribution. Due to the co-located performance anomalies, evidential correlation is higher, whereas, when performance anomalies are randomly distributed, we have more isolated evidences. We have also demonstrated these explicit evidential correlation (explained in SectionIV-A) results in networks



(a) DR with Random Loss Dist. (b) FPR with Random Loss Dist. (c) DR with Differential Loss Dist.

(d) FPR with Differential Loss Dist. (e) CCDF w.r.t Loss Distribution (f) Anomalous network Fraction

Fig. 3. Figures 3(a), 3(b), 3(c) and 3(d) show the effect of increase in topology size on performance with 15 performance anomalies and without considering positive evidences. Figure 3(e) shows cumulative CDF of evidence correlation under Random and Differential loss distributions, *PR* is participation ratio. Figure 3(f) shows effect of increase in fraction of network dropping packets on performance at fixed participation ratio with only negative evidences.

with both differential and random loss distribution models, in Figure 3(e). As we can see in Figure 3(e), evidential correlation is 10-20% higher in networks with differential loss distribution than networks with random distribution. Both of these performance scalability and evidential correlation results in Figure 3 comply each other.

4) Effect of Increase in Number of Errors on Performance Scalability: In Figure 3(f) we show the effect of change in fraction of network components acting as anomalous with different topology sizes, keeping a fixed participation ratio, *i.e.*, 30%. We have varied the fraction of network components acting anomalous between 5-35%. Figure 3(f) shows that when fraction of network components being anomalous exceeds 25%, the detection rate reaches below 80% and false positive rate reaches to almost 20%. It is because at fixed participation more anomalies results into more overlapping between evidences and if evidences are completely overlapped, isolating anomalies becomes very difficult. For example, if two evidences e_1 and e_2 are completely overlapped, $e_1 \Leftrightarrow e_2 = \{c_1, c_2, c_3, c_4\}$, then it is not possible to accurately diagnose and localize performance anomalies. In reality, normally this is not the case and mostly only a small fraction of network components act as performance anomalous at one time.

V. Conclusion

As overlay networks have extended the native infrastructure, the potential of performance problems in overlays have also increased beyond traditional networks. For efficient service management of overlay networks, in this paper, we have proposed a novel diagnosis technique, that only requires negative evidences from the end-users to localize performance anomalies and determine the packet loss contribution for each network component, without any active probing or sensor deployment in the network. We have formulated a constraint-satisfaction problem, that identifies a set components that cover all negative evidences and collectively explain all reported packet loss distribution. Our evaluation have demonstrated that the detection rate reaches to almost 90% for just 30% end-user participation and it is resilient to amount of spurious observations if this amount stays below 20%. Our solution is also scalable to a network topology of size 1500 nodes with greater than 90% accuracy and less than 10% false positive rate.

REFERENCES

[1] E. Al-Shaer, W. Marrero, and A. El-Atawy, "network configuration in a box: Towards end-to-end verification of network reachability and security." in *IEEE International Conference of Network Protocols (ICNP'2009)*, October 2009.

[2] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "Resilient overlay networks," in *Proceedings of the 18th ACM SOSP*, October 2001.

[3] "Planetlab," http://www.planet-lab.org.

[4] V. N. Padmanabhan, L. Qiu, and H. J. Wang, "Server-based inference of internet link lossiness," in *INFOCOM*, 2003.

[5] S. Agrawal, K. Naidu, and R. Rastogi, "Diagnosing link-level anomalies using passive probes," in *INFOCOM*, 2007.

[6] Y. Tang, E. Al-Shaer, and R. Boutaba, "Efficient fault localization using incremental alarm correlation and active investigation for internet and overlay networks," *IEEE Transactions on Network and Service Management (TNSM)*, 2008.

[7] G. J. Lee and L. Poole, "Diagnosis of tcp overlay connection failures using bayesian networks," in *SIGCOMM 06 Workshops on Mining network data*, September 2006.

[8] A. Batsakis, T. Malik, and A. Terzis, "Practical passive lossy link inference," in *Passive and Active Measurement Workshop*, 2005.

[9] Z. Duan, Z.-L. Zhang, and Y. T. Hou, "Service overlay networks: Slas, qos, and bandwidth provisioning." in *IEEE/ACM Trans. Netw.*, vol. 11, 2003, pp. 870–883.

[10] Y. Chen, D. Bindel, H. Song, and R. H. Katz, "An algebraic approach to practical and scalable overlay network monitoring," in *In Proceeding of ACM SIGCOMM*, 2004.

[11] K. Jung, Y. Lu, D. Shah, M. Sharma, and M. S. Squillante, "Revisiting stochastic loss networks: Structures and algorithms." in *ACM SIGMETRICS*, June 2008.

[12] N. G. Duffield, "Network tomography of binary network performance characteristics," in *IEEE Transactions on Information Theory*, vol. 52, 2006, pp. 5373–5388.

[13] M. Zhang, C. Zhang, V. Pai, L. Peterson, and R. Wang, "Planetseer: Internet path failure monitoring and characterization in wide-area services," in *OSDI*, 2004.

[14] K. Lakshminarayanan, I. Stoica, and S. Shenker, "Building a flexible and efficient routing infrastructure: Need and challenges," University of California Berkeley, Technical Report, 2003.

[15] M. Steinder and A. S. Sethi, "Non-deterministic diagnosis of end-to-end service failures in a multi-layer communication system," in *IEEE Conference on Computer Communications & Networks (ICCCN)*, 2001.

[16] ——, "Probabilistic fault diagnosis in communication systems through incremental hypothesis updating," vol. 45, no. 4, July 2004, pp. 537–562.

[17] M. Steinder and A. Sethi, "Probabilistic fault localization in communication systems using belief networks," in *IEEE/ACM Transactions on Networking*, vol. 12, 2004, pp. 809–822.

[18] Y. Tang and E. Al-shaer, "Sharing end-user negative symptoms for improving overlay network dependability." in *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, June 2009.

[19] Y. Tang, E. Al-Shaer, and R. Boutaba, "Active integrated fault localization in communication networks," in *IEEE/IFIP Symposium on Integrated Network Management*, 2005.

[20] "Z3 theorm prover." [Online]. Available: http://research.microsoft.com/en-us/um/redmond/projects/z3/

[21] "Yices: An smt solver." [Online]. Available: http://yices.csl.sri.com/

[22] L. D. Moura and N. Bjrner, *Satisfiability Modulo Theories: Introduction and Applications.* CACM, 2011.

[23] V. Pappas, B. Zhang, A. Terzis, and L. Zhang, "Fault-tolerant data delivery for multicast overlay networks," in *IEEE ICDCS*, March 2004.

[24] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker, "On the constancy of internet path properties," in *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, November 2001.

[25] A. Collins, R. S. Michalski, A. C. Theory, A. Collins, B. Beranek, , and N. Inc, *The logic of plausible reasoning. A Core Theory.* Cognitive Science, 1989.

[26] D. Lin and R. Morris, "Dynamics of random early detection." in *ACM SIGCOMM '97*, October 1997.

[27] "Congestion avoidance overview." [Online]. Available: http://www.cisco.com/en/US/docs/ios/12_0/qos/configuration/guide/qcconavd.html

[28] R. P. Srivastava, "Decision making under ambiguity: A belief-function perspective," in *Archives of Control Sciences*, vol. 6, 1997, pp. 5–27.

[29] "Brite topology generator," http://www.cs.bu.edu/brite/.