Towards Efficient Large-Scale VPN Monitoring and Diagnosis under Operational Constraints

Yao Zhao, Zhaosheng Zhu, Yan Chen
Northwestern University
{yzhao,zzh321,ychen}@cs.northwestern.edu

Dan Pei, Jia Wang
AT&T Labs – Research
{peidan,jiawang}@research.att.com

Abstract— **Continuous monitoring and diagnosis of network performance are of crucial importance for the Internet access service and virtual private network (VPN) service providers. Various operational constraints, which are crucial to the practice, are largely ignored in previous monitoring system designs, or are simply replaced with load balancing problems which do not work for real heterogeneous networks.**

Given these real-world challenges, in this paper, we design a $VScope$ monitoring system with the following contributions. First, we design a greedy-assisted linear programming algorithm to select as few monitors as possible that can monitor the whole network under the operational constraints. Secondly, VScope takes a *multi-round* measurement approach to further reduce monitors deployment/management cost, by scheduling the path measurements in different rounds under the operational constraints. Evaluations based on several real VPN topologies from a tier-1 ISP as well as some other synthetic topologies demonstrate that VScope is promising to solve the aforementioned challenges.

1. Introduction

Recently the Internet has witnessed an unprecedented growth in terms of the scale of its infrastructure, the traffic load, as well as the abundant applications. More importantly, there is an exponential growth for MPLS-based IP Virtual Private Networks (VPN) recently. Large enterprise networks often have multiple sites that are at separate geographical locations. For example, large corporations such as IBM and Nokia have offices/branches that locate in many countries, and large retail stores such as Macys and Wal-Mart have thousands of stores globally. To connect sites (*e.g.*, offices or stores) within an enterprise network, instead of deploying/leasing physical lines between sites, they usually let ISPs provide and manage the connectivity via MPLS/VPN. This approach has been adopted widely because of its low cost and great flexibility. Because a VPN provider is often the sole provider of connectivity among a customer's sites, continuous monitoring and diagnosis of VPN performance are of crucial importance for the VPN service providers to ensure the reliability and quality of service.

Today, ISPs heavily rely on the standard passive monitoring approach via SNMP, which usually polls the status of each router/switch periodically. However, there are several issues. First, an ISP usually provides VPN services to a large number of customers such as enterprise networks, all of which run on top of the same ISP infrastructure. As such, the ISP needs to monitor hundreds of thousands of routers. Therefore, it is infeasible to frequently poll every router due to the large bandwidth and management overhead. Secondly, SNMP based monitoring is unable to measure the path-level features such as latency.

Therefore, active measurements are important complement to the SNMP based monitoring approach and are also used by ISPs widely. However, most existing network monitoring and diagnosis designs [1]–[7] miss an important piece: various constraints that should be imposed on the monitors and links so that the measurement does not interfere with the normal operation or traffic, and meets the business requirement. For example, the capacity of access links that connects each site belong to a single VPN can be very limited, *e.g.*, only 1.54Mbps as we observed from the majority of access links in thousands of VPNs managed by a tier-1 ISP. This is because customers often do not have incentive to pay for their providers to over provision the access link capacity. We define the *operational constraints* to be the set of constraints or rules that the monitoring system should comply to. For example, a typical constraint can be that all the measurement overhead over a link cannot exceed 1% of the link capacity. Thus when selecting the monitors or paths for monitoring without considering these constraints (as in [1, 4, 5, 7]), it is very likely to severely overload some monitors and/or links.

In this paper, our goal is to design a monitoring and diagnosis system for the VPN infrastructure that ISPs deploy to host VPN services. Taking the operational constraints into account makes this problem very challenging and unique from the existing work for the following reasons.

- The measurement design problem is not only an optimization problem, but also a constraint satisfactory problem. For example, minimizing the number of monitors or scheduling paths to measure under the constraints become harder than some notorious NP-hard problems.
- Most tomography work assumes that all the paths to be monitored will be measured simultaneously [1, 4, 5, 7]. However, this setup may not be true or efficient under the real-world constraints.

To address these challenges, we propose $VScope$, a $continuous$ monitoring and diagnosis system for VPN. While we mainly focus on VPN service in this paper, VScope is general enough to work on any other network whose resources are limited and the operational constraints should be considered in its active monitoring system (*e.g.*, IP network of a small Tier-3 ISP). The key idea is to select the candidate routers as monitors and schedules the paths to be measured by the monitors *in multiple rounds*. This is the monitor setup phase of VScope. In the second phase, VScope continuously monitors the networks and locates the congested links for diagnosis. Such a multi-round measurement approach gives a smooth tradeoff between measurement frequency and monitors deployment/management cost. In particular, we make the following contributions in designing the VScope.

First, we design algorithms to select as few monitors as possible that can monitor the whole network under the operational constraints. The special case of our problem ignoring the operational constraints is shown to be NP-hard in [1]. Considering the operational constraints, we model our problem as a unique combination of the two-level nested Set Cover problem and constraint satisfaction problem. We found that no existing solutions such as those for variants of Set Cover problem [8] can be directly applied to solve this new problem. Thus we design a greedy-assisted linear programming algorithm for it. In addition, we develop a simple but scalable greedy algorithm for a smooth efficiency-optimality tradeoff.

Secondly, with the single-round measurement algorithms as the basis, we propose three algorithms to schedule the path measurements in different rounds obeying the operational constraints. Both analytical and experimental evaluations demonstrate that we can effectively approximate the optimal solutions with little constraint violation.

Besides some synthetic topologies, we mainly evaluate the VScope system with one IP network topology and two VPN topologies, *all with the real topologies, capacities and constraints*, from a tier-1 ISP. The sizes of networks vary from hundreds to hundreds of thousands of routers. The results demonstrate that our multi-round approach can significantly reduce the number of routers for monitors to only about 5% of all routers when covering all the links with all the constraints.

The rest of the paper is organized as follows. We introduce the problem and VScope architecture in Section 2. We present our design on monitor selection in Section 3. The dynamics issues are discussed in Section 4. Then we show the evaluation methodology and results in Section 5. Finally, we present related work in Section 6 and conclude in Section 7.

2. Problem Definition and Architecture

1. Problem Definition

From the ISP operational perspective, the goals of network monitoring are two-fold. First, ISPs need to actively measure or infer the performance of all the possible paths through the VPN. Second, ISPs also need to quickly identify the root cause of performance degradation or service disruption. The monitoring problem can be divided into two phases: the setup phase for monitor selection and the continuous monitoring and fault diagnosis phase. In this section, we define each of the subproblems in terms of these two phases.

1) Background on ISP VPN Infrastructure: A layer-3 Virtual Private Network (VPN) refers to a set of sites among which communication takes place over a shared network infrastructure called a *VPN backbone*. Figure 1 shows a VPN backbone with two VPNs and three sites. *Customer Edge device routers (CE routers)* are connected to the *Provider Edge device routers (PE routers)* in the provider network via external BGP (eBGP). Other routers in the provider network are called *Provider's device routers (P routers)*. Each PE router maintains a Virtual Routing and Forwarding (VRF) table for each VPN so that routes from different VPN customers remain distinct and separate even if multiple VPN customers use the same IP address space. Internal BGP (iBGP) is used to distribute the VPN routes

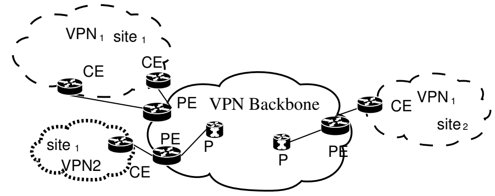

Fig. 1. Example of Layer-3 IP VPN infrastructure.

within the VPN backbone. Within the VPN backbone, *Multi-Protocol Label Switching (MPLS)* tunnels between PEs are used to forward packets. It is worth mention that the goal of the VScope system is to monitor and diagnose the whole ISP *VPN infrastructure* including the shared VPN backbone and the customer routers, instead of a single VPN.

2) Measurement Constraints: One guideline for active measurements is to avoid interrupting the normal network traffic or overloading network or computation resources. After consulting network operators of a major tier-1 ISP, we consider the following realistic measurement constraints:

- **Monitor constraints.** We define the routers (*e.g.* PE,CE and P routers) that can be monitors as *candidate routers*. Some routers cannot be selected as monitors for various business and hardware reasons. For example, some CE routers are not managed by the VPN provider. More importantly, each candidate monitor has limited probing ability (*e.g.*, 50 probes/second). Given a fixed measurement overhead on each measured path, a monitor thus can measure only a limited number of paths simultaneously.
- **Replier constraints.** The routers that can reply to the probes from the monitors are *repliers*. To avoid overloading the replier routers, we enforce the replier constraint, which specifies the number of probes that the replier can reply in a certain period. Note the operators may need to adjust the access list and rate limit of the router configuration to comply with the replier constraint without introducing security holes. For example, a router can be configured to allow 100 ICMP Echo Reply per second from the senders in some IP prefix.
- **Link constraints.** Every link has its own bandwidth. The measurement overhead on a link should not exceed a certain portion of the link bandwidth (*e.g.*, 1%). Generally, the link capacity in the backbone networks is pretty large, while the edge links usually have much lower capacity. For example, among thousands enterprise VPN configurations that we have examined, more than 70% access links have capacity of only 1.54 Mbps, while the backbone links usually have capacity of 150 Mbps or more. Considering there are many more access links than backbone links, we can see that most of the links have low bandwidth.
- **Measurement path selection constraints.** VPN provides the traffic isolation between different customers. Only the sites/routers within the same VPN can communicate with each other. The path selected for measurement in VScope needs to satisfy this constraint too. Note the measured paths are round-trip paths because the non-monitor routers can only reply to probes.

3) Monitor Setup Phase: Generally, an active network monitor and diagnosis system needs to select some monitors as well as path sets to be monitored. In VScope, one goal is to

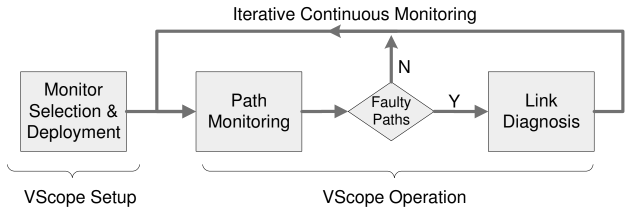

Fig. 2. VScope System Architecture.

minimize the number of monitors to save the installation and management cost. Meanwhile, the monitored path sets should cover all the links in the network, as in other related works [1, 9]. However, our VScope system design is unique compared to previous work due to the consideration of operational constraints, which is critical to VPN and greatly complicates the problem. Particularly, in previous works [1, 4, 5, 7, 9] all the selected paths are measured simultaneously because there are no constraints on the abilities of the routers and links in the model. However, given the operational constraints, we find that scheduling path measurements in multiple rounds is an efficient and necessary approach to save on monitor installation cost. Therefore, the constrained scheduling problem becomes a unique problem in our VScope system.

Mathematically, the monitor selection problem can be abstracted and generalized as follows: Let $G(V, E, P)$ be a network where V is the vertex set, E is the edge set and P is the predefined set of paths. Assume Φ is a set of rules that determines if the selection of paths $P' \subset P$ is allowed or not. The problem is to select a path set P^* satisfying Φ and for each edge $e \in E$ there exists a path $p \in P^*$ with $e \in p$. Meanwhile, let V^* be the set of starting vertices of all paths in P^*, and the goal is to minimize the size of V^*.

4) Monitoring and Fault Diagnosis Phase: VScope monitoring involves periodically probing or inferring the path performance metrics, such as reachability, latency, loss rate, and so on. Locating faulty links from path measurements is a hard problem and a lot of algorithms [2, 6, 7] have already been designed for this purpose. Our VScope system leverages on and extends the existing approaches [6], but this is not our focus. Especially given the space limit, we only define the monitoring and fault diagnosis problem in this paper for completeness, but leave all the details of the algorithms and evaluations in our technique report [10]. Specifically, we consider the following problem in our VScope system:

When faulty paths are discovered in the path monitoring phase, how can we quickly select some paths under the *operational constraints* to be further measured so that the faulty link(s) can be accurately identified?

2. Architecture

Figure 2 shows the architecture of our system. The architecture has two components: *monitor selection*, and *continuous monitoring and diagnosis*. First, a set of monitors are selected according to the algorithms introduced in Section 3, and measurement boxes/software are installed. Then the monitors probe paths and diagnose faulty links periodically. In each round, a set of paths is measured using active probing. Next, if some paths are found to be faulty, the diagnosis component will further locate the faulty links along the faulty paths. Additional path measurements are selected and conducted for this purpose under the operation constraints (details in [10]). VScope has a centralized coordinator, like the network operation centers for many major ISPs, which assigns measurement tasks to monitors, collects the measurement results, and detects faulty paths and identifies faulty links.

3. VSCOPE MONITOR SELECTION

As described in Section 2.1.3, the goal of the monitor selection is to select minimal number of monitors to actively monitor all links in the network under the operation constraints.

The constraint satisfactory problems including our problem is usually NP-hard (See Section 3.2 for the hardness of our problem), and even the best algorithms may not be able to achieve the satisfaction [11]. In our VScope system, we do not plan to struggle with the notorious satisfaction problem. Instead, we propose to schedule the path measurements into different rounds [1] to "reduce" the harsh constraints so that simple algorithms like the greedy algorithm can at least find a solution easily. Meanwhile we find multi-round can significantly cut down the number of monitors required to monitor the networks.

1. Overview of Multi-round Monitoring

The main idea of our multi-round monitoring is as follows: we consider R rounds of back-to-back measurements and in each measurement round different paths are measured by the selected monitors. Finally, every link is covered by at least one of the R rounds of measurements. The multi-round monitor selection algorithm tries to minimize the number of monitors that can cover all the links in a certain number of rounds (R).

An optimal solution should consider both the monitor/path selection and the schedule of the path measurements in multiple rounds at the same time, which is very hard involving the both monitor/path selection and scheduling problems. Therefore, we propose a two-step solution for the multi-round monitor selection problem. First we convert the multi-round selection problem to the "single-round" selection problem by relaxing the monitor's constraints and link bandwidth constraints by a factor of the round number R. In this step, we obtain the selected monitors as well as paths to be measured. In the second step, we schedule the paths to be measured in the R rounds appropriately, trying to satisfying the constraints of each round.

2. Monitor Selection

The monitor selection problem seems to be similar to the problem in [1], which is a simpler case of our problem without considering the operation constraints. And in [1] Bejerano *et al.* proved that this simplified case of our problem is NP-hard. The monitor selection problem resembles the well-known Minimum Set Cover problem [8, p. 118]. One can imagine each link as an element and each candidate router as corresponding to a set. We say a path *covers* a link if the link is on the path, and a link is *associated with* a router if the link is covered in at least one of the paths starting from the router. Hence a router's corresponding *set* contains all the links associated with the router. The Minimal Set Cover problem involves finding the smallest number of sets (or routers) that cover all the elements (or links). However, the existence of monitor/replier/link

[1] Paths in the same round are measured simultaneously.

Symbols	Meaning
N	Number of routers
S	Number of links
P_{ij}	The path from router i to router j
L_k	The kth link. $L_k \in P_{ij}$ if this link on path P_{ij}
x_i	1, if node i is a monitor, otherwise 0
y_{ij}	1, if path P_{ij} is measured, otherwise 0
z_k	1, if link k is covered, otherwise 0
c_i	The number of paths that node i can measure
r_i	The number of paths that node i can reply
b_k	Max # of measured paths that can pass link k
OPT	# of monitors required in the best solution

TABLE I Notation used in the paper

constraints makes our problem first a constraint satisfactory problem.

Given complicated constraints, the classic approximation algorithms for the Set Cover problem and its variants [8] can not be directly applied to solve our problem. While in principle we still use the classic algorithms of approximation algorithm (*e.g.*, greedy algorithm and linear programming), there are substantial challenges to realize the algorithms for our realistic problem. Next, we present two algorithms, the greedy algorithm and the linear programming with random rounding algorithm to solve our monitor selection problem. Table I illustrates the notations used in the paper.

1 Let $L = \{l_1, l_2, \ldots, l_S\}$ be the set of links;
2 Let $C = \{r_1, r_2, \ldots, r_N\}$ be the set of candidate routers;
3 Let $T = \emptyset$ be the initial set of covered links;
4 Let $R = \emptyset$ be the output of selected monitors;
5 **while** $L - T \neq \emptyset$ **do**
6 $\quad S^* = \emptyset$ and $r = \emptyset$;
7 $\quad$ **foreach** $r_i \in C - R$ **do**
8 $\quad\quad$ Select the path set S_i which covers the maximum number of the links in $L-T$ under link constraints;
9 $\quad\quad$ **if** $|S_i| > |S^*|$ **then**
10 $\quad\quad\quad S^* = S_i$, $r = r_i$;
$\quad$ **end**
11 $\quad R = R \cup \{r\}$, $T = T \cup S^*$;
12 $\quad$ Update the constraints of links;
end

Algorithm 1: Greedy algorithm for monitor selection.

1) Greedy Monitor Selection Algorithm: Greedy algorithms are usually one of the most straightforward and to deal with some NP-hard problems. Especially in Minimum Set Cover problem, pure greedy algorithm turns out to be a $\log M$-approximation algorithm, where M is the number of elements to cover [8]. Besides, in the average case, greedy algorithm is much more efficient than what the theoretic bound says.

In this section, we introduce a simple greedy algorithm inspired by the greedy algorithm for Minimum Set Cover problem. Our monitor selection problem looks like a two-level nested Minimum Set Cover problem and Maximum k-Coverage problem [12] to some extent. Algorithm 1 describes the greedy algorithm for monitor selection. The basic idea is to greedily select one router at a time, which can monitor the largest number of links that have not been covered yet.

However, the problem of evaluating the gain of adding a router as a monitor is a variant of Maximum k-Coverage problem, an NP-hard problem [12]. The Maximum k-Coverage problem is to select k sets from certain candidate sets so that the maximum elements are covered in the union of the selected sets. Considering the paths as sets and links as elements, it is a k-Coverage problem to find out the number of links covered by a fixed number of paths that a router can simultaneously monitor, if we do not consider link bandwidth constraints. Similarly, our greedy algorithm also selects iteratively the path that can cover most new links while complying with the link constraints. Because of space limit, line 8 in Algorithm 1 omits the details.

It is worth mention that Algorithm 1 degenerates to be the simpler greedy algorithm in [1] if we ignore all the link constraints and monitor constraints. Step 8 in Algorithm 1 turns out to select all the path starting from router r_i, or the so called routing tree in [1].

2) Linear Programming based Monitor Selection Algorithm:

1) Integer Linear Programming: We first formulate our monitor minimization problem as an integer linear programming problem (ILP) as follows (See Table I for notations):

$$P: \quad \text{Minimize} \quad \sum_i x_i \tag{1}$$
$$s.t. \quad y_{ij} \leq x_i, \ \forall i, \ \forall j \tag{2}$$
$$\sum_j y_{ij} \leq c_i \cdot x_i, \ \forall i \tag{3}$$
$$\sum_j y_{ji} \leq r_i, \ \forall i \tag{4}$$
$$\sum_{\forall i, \ \forall j, \ L_k \in P_{ij}} y_{ij} \geq 1, \ \forall k \tag{5}$$
$$\sum_{\forall i, \ \forall j, \ L_k \in P_{ij}} y_{ij} \leq b_k, \ \forall k \tag{6}$$

Formula 1 is the minimization goal of the ILP, *i.e.*, minimizing the number of monitors needed. Inequality (2) means a path can be measured if and only if the source router of the path is selected as a monitor. The monitor and replier constraints are formulated in Inequality (3) and (4). Inequality (5) shows that a link is covered when at least one of the paths containing the link is selected. Link bandwidth constraint is enforced by Inequality (6).

2) Relaxed Linear Programming: Integer linear programming is a NP-Complete problem [13], and thus solving it may not be feasible. We use the classic relaxation techniques to relax the $\{0,1\}$-ILP to a normal linear programming problems and then apply the random rounding scheme to achieve the optimality bound in terms of statistical expectation. To relax the integer linear programming, we simply add the following constraints and remove the $\{0,1\}$-solution requirement:

$$0 \leq x_i \leq 1, \quad 0 \leq y_{ij} \leq 1, \quad \forall i, \ \forall j$$

After relaxation both x and y are real numbers in the range [0,1], and the linear programming problem can be solved in polynomial time. Suppose the solution is x_i^*, y_{ij}^*. We do the random rounding in the following way:

$$X_i = \begin{cases} 1 & \text{with probability} \quad x_i^* \\ 0 & \text{with probability} \quad 1 - x_i^* \end{cases} \tag{7}$$

$$Y_{ij} = \begin{cases} 1 & \text{with probability } y_{ij}^*/x_i^*, \text{ if } X_i = 1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

If X_i is rounded to 1, the corresponding router is selected as a monitor. Once a router is selected as a monitor, the paths starting from the router have some chance to be selected to measure with the probability y_{ij}^*/x_i^*. Then the value of z_k, *i.e.* whether a link is covered or not, is decided by the rounded Y_{ij}. Let random variables $X = \sum_i X_i$ and $Z = \sum_k z_k$. We have the following theorem:

Theorem 1: After applying random rounding to the solutions of the LP problem of the monitor selection, $E(X) \leq OPT$, and $E(Y_{ij}) = y_{ij}^*$.

The proof of Theorem 1 can be simply proved using the basic probability theory and we omit the details because of space limit. Theorem 1 shows that in expectation we select no more than OPT monitors. However, after rounding not all the links are covered. Note that in the standard LP algorithm for Minimum Set Cover problem, several random rounding results are combined together to obtain the 100% coverage of all the links. In our monitor selection problem, simply combining multiple results of random rounding will violate the monitor constraints and link bandwidth limitations. Therefore, we combine the LP-based algorithm with the greedy algorithm introduced in Section 3.2.1 to achieve 100% link coverage.

We apply the following Theorem 2 [14] to show that with pretty large probability, the random rounding results are not much larger than the expected results.

Theorem 2: Let V be the sum of independent $\{0, 1\}$ random variables, and $\mu > 0$ be the expected value of V. Then for $\forall \epsilon > 0$,

$$P_r(V \geq (1+\epsilon)\mu) < e^{-\mu \min\{\epsilon,\epsilon^2\}/3}.$$

For example, let $\mu = 12$ and $\epsilon = 1$, then $P_r(V > 24) < 0.018$. According to Theorem 2, we can see that the probability of large violation of the monitor constraint and link constraint is small. For example, inequality 3 enforces the monitor constraint in the linear programming and after random rounding we have $E[\sum_j Y_{ij}] \leq \sum_j y_{ij}^* \leq c_i$. In our setup, usually one monitor can measure 12 paths simultaneously (*i.e.*, $c_i = 12$), hence we have $P_r(\sum_j Y_{ij} > 2c_i) < 0.018$. To further reduce this violation, we can run random rounding several times to find the one which has minimal violations. The result shows that there are no violations to the constraints in our experiments on real topologies (See Section 5.2).

3) Greedy-assisted Relaxed Linear Programming: We take the LP results as a good starting point, which selects a certain number of monitors and paths associated with the monitors already. After removing the already covered links, we continue to use the greedy algorithm to add more and more monitors until all the links are covered. The algorithm is also called *LP+Greedy* in short.

Although it is hard to prove the bound for the greedy-assisted LP algorithm, we expect it to be more efficient compared to the pure greedy algorithm because of the good starting point. As shown in our experimental results (See Section 5), this hybrid approach is better than the pure greedy algorithm in terms of minimizing the number of monitors. Additionally, the greedy algorithm sometimes fails to select monitors that cover all the links under the operational constraints simply because it does not try to balance the loads on nodes and links.

3. Multi-round Path Scheduling

We now introduce the path scheduling algorithm. It is worth mentioning that the path scheduling problem itself is also an NP-hard problem. We can reduce the well-known minimum graph coloring problem (which is NP-hard [15]) to our path scheduling problem. One can imagine a round as a color, a path as a vertex and let two paths share a link if the corresponding vertices have an edge. We omit the detailed proof for space limit. In this paper, we propose an integer linear programming (ILP) with relaxation to solve the scheduling problem. Meanwhile, we also include two other straightforward and simpler scheduling algorithms for comparison, a simple randomized algorithm and a greedy algorithm. The simple randomized algorithm and the ILP-based algorithm have nice theoretical stochastic bounds on the results, and the greedy algorithm clearly has the optimization goal as the ILP-based algorithm. Although theoretically we cannot prove the ILP-based algorithm with relaxation is the best of the three, our simulation results on practical scenarios shows the advantages of the ILP-based algorithm.

Note that monitor constraints are easy to satisfy because monitors are independent in terms of the monitor constraints. However in some extreme cases, there may be some link constraint violations in some rounds even if we have the optimal scheduling algorithm. Therefore, in such cases our scheduling algorithm tries to minimize the constraint violations. We define the link violation degree of a link as $\frac{n}{b} - 1(n > b)$ where n is the scheduled number of paths over the link and b is the link constraint of the link. We consider two metrics that quantify the violation degree: 1) maximum link violation degree (MLVD); 2) average link violation degree (ALVD).

1) Simple Randomized Algorithms: For any path p to be measured, we simply randomly select a round of the R rounds and schedule to measure the path p in this round. In the sense of expectation, the randomized scheduling results comply with the monitor constraints and link bandwidth constraints in each round. For example, the monitor i will monitor no more than $N \times c_i$ paths in total, hence in every round at most c_i paths from the monitor i are expected to be measured. However, for example, in a randomized instance, a monitor may monitor paths more than expected and hence the monitor constraint is violated. Similarly, we can apply Theorem 2 to quantify the violation degree and possibility for monitor constraints and link constraints.

2) Greedy Algorithm: The second algorithm we also consider is a greedy algorithm. Basically, the greedy algorithm adds paths to the possible rounds of measurement, trying to minimize the violations of the system's constraints. It is easy for a greedy algorithm to schedule the path measurement so that monitor's constraints are all satisfied. However, link constraint violations may happen in some cases. Therefore, we let the object function of our greedy algorithm to minimize the maximum link violation degree or the average link violation degree of all the links. In each step, the greedy algorithm picks a path in the measurement set and put the path to a certain round so that monitor constraints are not violated and the maximum (or average) link violation degree so far is minimized.

3) LP based Randomized Algorithm: The last algorithm we propose is to use integer linear programming first, and then use the relaxation and random rounding algorithm described in Section 3.2.2 to convert it to linear programming. The objective function is minimizing the maximum link violation degree or the average link violation degree, which is the same as the greedy algorithm (See Section 3.3.2). Let $y_{ijr} = 1$ if path P_{ij} is scheduled to be measured in round r, and $y_{ijr} = 0$ otherwise. The integer linear programming is formulated to minimize the maximum link violation degree:

$$\begin{array}{lll} P: & \text{Minimize} & v \\ & s.t. & \sum_r y_{ijr} = 1, \ \forall i, j \\ & & \sum_j y_{ijr} \le c_i, \ \forall i \\ & & \sum_{\forall i, \ \forall j, \ L_k \in P_{ij}} y_{ijr} - b_k \le v \times b_k, \ \forall k, r \\ & & y_{ijr} \in \{0, 1\} \end{array} \tag{9}$$

Minimizing the average link violation degree is very similar so we omit the formula for the interest of space. Also we can apply Theorem 2 to quantify the violation degree and possibility for monitor constraints and link constraints after random rounding.

4. Robustness and Adaptivity in Dynamic Scenarios

In previous sections, we have assumed that the network topology and routing are static. In reality, the networks are dynamic with the changes of routers, links and so on. Therefore, our VScope system needs to be robust against the temporary or permanent changes, and be adaptive to the dynamics in the network. First, in VScope we consider the redundancy in monitor selection to obtain the robustness. Simple modification in the LP based algorithm and greedy monitor selection algorithm will introduce redundancy in the system, *e.g.*, requiring each link to be covered by multiple paths to handle routing changes. Second, we also propose the incremental path reselection algorithm to reduce the redistribution overhead of the monitoring jobs to monitors. However, because of the space limit, we do not describe the details of these algorithms in this paper and the details can be found in [10].

5. Evaluation

In this section, we will first describe the evaluation methodology. Then we present the results of the baseline monitor selection, multi-round monitor selection, and path scheduling. Finally we show the computation speed results.

1. Evaluation Methodology

1) Topology Dataset: We evaluate our VScope over various synthetic and real topologies, not limited to VPN topologies only, because VPN services are growing fast and future VPN infrastructure may have quite different topologies. The synthetic topologies we use are generated by BRITE [16] with the Barabasi-Albert model and Waxman model. The four real topologies are from a tier-1 ISP. The smallest one is a VPN backbone in US (named *VB* in the rest of the paper) and the second topology is an ISP IP network topology, called *IP-EX*. These first two networks have relative large bandwidth compared to the real VPN infrastructure. The two VPN infrastructure topologies are *V1-EX* and *V2-EX*, respectively. Table II gives the orders of magnitude for the number of routers, links and VPNs in these topologies. For the space limit, we only present the evaluation results with the four real topologies, and omit the similar results of synthetic topologies.

Statistics	V1-EX	V2-EX	VB	IP-EX
# of PE routers	100s	100s	100s	100s
# of P routers	100s	100s	100s	100s
# of CE routers	100000s	10000s	N/A	10000s
# of Links	100000s	10000s	1000s	10000s
# of VPNs	1000s	1000s	N/A	N/A

TABLE II Statistics of the VPN and IP Topologies.

Number of paths a monitor can measure	12/round
Number of paths a replier can respond	24/round
Packet probing rates per path	4 pkt/s
Bandwidth consumed by each path measurement	1.6 Kbps
Percent of link bandwidth allowed for probing	1%

TABLE III Basic configuration and constraints.

Table III describes the basic configuration and constraints we select for the baseline experiments. We use them as the default setup unless specified otherwise. After consulting with the ISP management team, the rule of thumb is to have one monitor send about 3000 probes per minute, and usually the probing frequency of one path is four probes per second. We set our constraints accordingly, *e.g.*, we set the monitor constraint as 12 paths. This means the monitor can measure 12 paths simultaneously. The link capacity is very heterogeneous in VPN; For example, for V1-EX topology, a few backbone links have more than 150 Mbps capacity, but most link capacities are only 1.54 Mbps.

2) Evaluation Metrics: Our metrics include 1) the number of selected monitors in monitor setup phase; 2) maximum link violation degree or average link violation degree (See Section 3.3) in multi-round path scheduling ; 3) running speed of the algorithms for monitor setup.

Due to the anonymity requirement from the tier-1 ISP, we cannot provide the number of monitors or links in the studied topologies. So we only show the percentage of monitors selected and the percentage of links covered.

2. Baseline Monitor Selection Results

In this section, we present the results of the single-round monitor selection algorithms of both the LP+Greedy algorithm and the pure greedy algorithms. We first present the baseline experiment results with the VB backbone topology. And then we run more extensive experiments, varying the constraints and the topologies.

1) Results of Baseline Setup: We use the default configuration in Table III and run the two monitor selection algorithms (the LP+Greedy algorithm and pure greedy algorithm) on the VB topologies. The LP+Greedy algorithm selects about 13% candidate routers as monitors while the pure greedy algorithm selects 14% routers as monitors. And both algorithms can cover all the links in the network. In the default configuration, we can see that the LP+Greedy algorithm performs a little bit better than the pure greedy algorithm.

2) Varying Monitor Constraints: Intuitively under certain monitor and link bandwidth constraints, the monitor selection algorithm may not be able to achieve 100% link coverage. Fortunately in our simulations, the algorithms can always achieve full link coverage and hence we only need to consider the number of selected monitors.

Figure 3 shows the percentage of routers that are selected as monitors given different monitor constraints. Clearly, for the LP+Greedy algorithm, the higher monitor constraint, the fewer

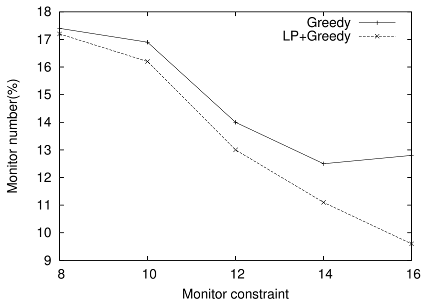

Fig. 3. Percentage of routers selected as monitors as a function of monitor constraints in VB.

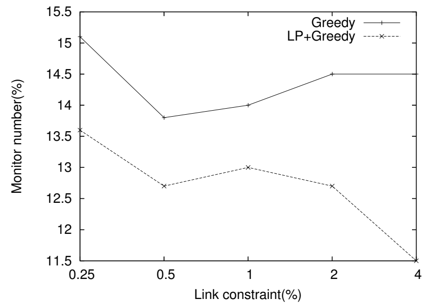

Fig. 4. Percentage of routers selected as monitors as a function of link constraints in VB.

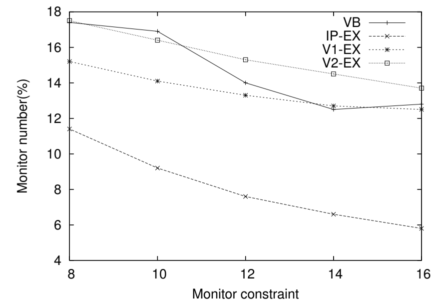

Fig. 5. Percentage of routers selected as monitors as a function of monitor constraints in 4 topologies.

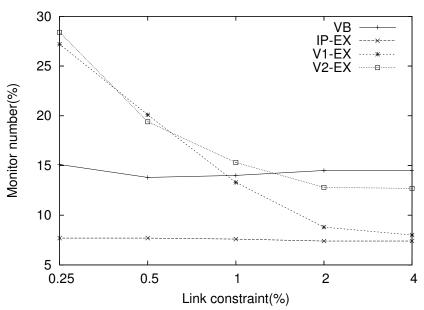

Fig. 6. Percentage of routers selected as monitors as a function of link constraints in 4 topologies.

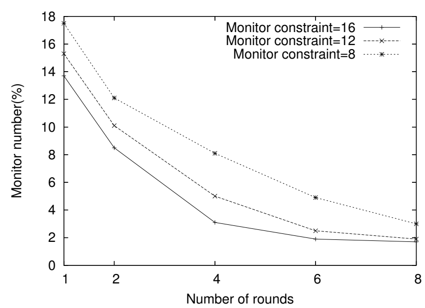

Fig. 7. Multi-round monitor selection with different monitor constraints.

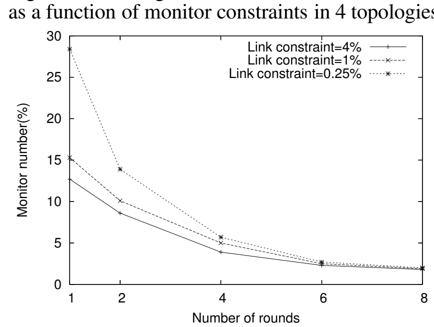

Fig. 8. Multi-round monitor selection with different link constraints.

monitors are required. However, there are some exceptions in the pure greedy algorithm. We believe this instability problem of the pure greedy algorithm lies in the nature of missing global optimization in the resource allocation. Overall, the LP+Greedy algorithm outperforms the pure greedy algorithm by selecting fewer monitors. In some cases, the greedy algorithm selects about 30% more monitors than the LP+Greedy algorithm (*e.g.* when a monitor can measure 16 paths simultaneously).

3) Varying Link Bandwidth Constraints: In this section, we vary the link bandwidth constraints with the VB topology in the simulation. Usually the more link bandwidth is allowed for measurement, the larger flexibility for monitors to select paths to measure.

Figure 4 demonstrates how many routers are selected as monitors by the two monitor selection algorithms. Again, we find the LP+Greedy algorithm is better than the pure greedy algorithm, as the latter always selects more monitors. For example, when link constraint is 4% of link capacity, the LP+Greedy algorithm selects about 25% less monitors than the pure greedy algorithm. Interestingly, looser link constraints do not always result in fewer monitors for both algorithms. Again, locally optimized feature of the greedy algorithm may play an important role for such results.

4) Varying Topologies: We present the monitor selection results on different topologies in the following paragraphs. Note we only show the result of the pure greedy monitor selection algorithm. The linear programming based algorithm cannot scale to the extremely large network topologies which have hundreds of thousands of nodes and hundreds of millions of paths.

Figure 5 shows the number of monitors selected in different topologies while varying the monitor constraint. As we expected, for all topologies the percentage of routers selected as monitors drops as each monitor can measure more paths. Meanwhile the dropping rates become flat as monitor constraints increase.

Figure 6 shows the effect of link bandwidth constraints on the monitor selection. In the V1-EX and V2-EX topologies, link bandwidth constraints play a very important role. For example, in the V1-EX topology, less than 15% routers are selected as monitors if 1% link bandwidth is used for measurement; while the percentage of monitors increases to about 27% when only we use 0.25% link bandwidths for measurement. On the contrary, the IP-EX topology may have large link bandwidth and the monitor selection is not affected by the link bandwidth constraints at all. Since the configurations of the ISP measurement are also flexible (*e.g.* changing the probe rate on a path to vary the monitor constraints), it is reasonable to select a practical constraint configuration to achieve a good tradeoff between the deployment cost and monitoring performance.

5) Summary: Even with the LP+Greedy algorithm which performs superior to the pure greedy one, the results show that the single round monitoring is inefficient for the number of monitors selected, suggesting that the multi-round monitoring is necessary in practice.

3. Multi-round Monitor Selection Results

In this section, we present the simulation results of the multi-round monitor selection algorithm and the three multi-round scheduling algorithms on the V1-EX topology and omit the similar results of other topologies. As described in Section 3.3, there can be two different optimization goals of the greedy and LP-based scheduling algorithm: minimizing the Maximum Link Violation Degree (MLVD) and minimizing the Average Link Violation Degree (ALVD). We present the simulation results of the both goals in the following simulations.

We simulate the three multi-round monitor selection algorithms under the baseline setup (See Table III) first, and then vary the configurations such as link bandwidth constraints. We

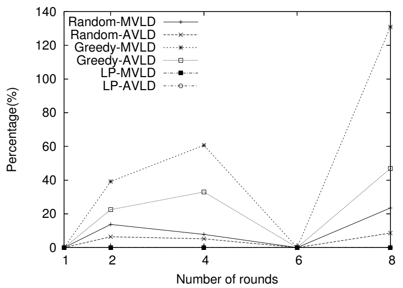

Fig. 9. Link violation degree of different algorithms.

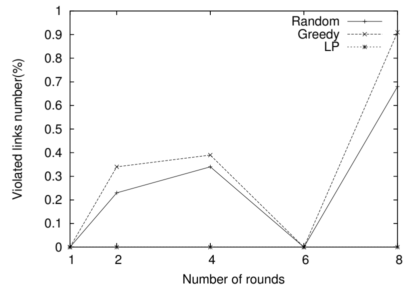

Fig. 10. Percentage of links with violation in different algorithms.

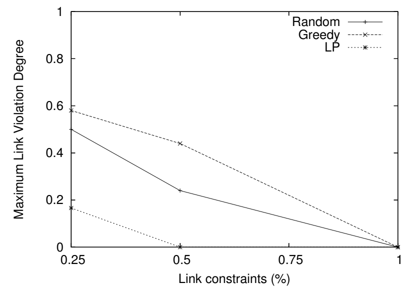

Fig. 11. Maximum link violation degree of different link constraints.

also vary the number of rounds from one to eight to show the efficiency of the multi-round monitor selection algorithm.

1) Monitor Selection Results: Figures 7 and 8 show the number of monitors selected under different simulation setups. Clearly, the percentage of routers selected as monitors decreases as the number of rounds increases. For example, in the baseline setup (*i.e.*, monitor constraint is 12), with round number as four we select only 6.2% routers as monitors, which is half of that selected by the single-round algorithm. However, Figures 7 and 8 also show that more rounds do not save many monitors when the number of rounds is more than four. Actually, the multi-round approach is a way of relaxing the constraints of the monitoring, and there is a minimum number of required monitors even without any constraints. In our topologies, we find the round number of four is a good trade-off between the cost of monitors (*i.e.*, number of monitors) and the measurement frequency in the current topologies and constraints.

2) Multi-round Scheduling Algorithm Results:

1) Comparing different scheduling algorithms: We first compare the three scheduling algorithms, simple random algorithm, greedy algorithm and LP-based algorithm using the maximum link violation degree as the optimization goal. Note in the baseline setup, link violation is always zero for all the three algorithms, so we show the comparison results under a tighter constraint setup for comparison where only 0.25% link bandwidth can be used for measurements.

Figure 9 shows the maximum link violation degree (MLVD) and average link violation degree (ALVD) of the three algorithms while varying the number of rounds. Clearly, LP-based algorithm works the best, as it always has no violation in every setup. Surprisingly, simple random algorithm outperforms the greedy algorithm. Note for the simple random algorithm, we run the algorithm with different random seeds for several times and pick the best randomized result. So this suggests that randomization is quite helpful in our cases, while the simple greedy algorithm may be far from global optimization. Figure 10 shows percentage of links that link constraint violation happen after scheduling. The figure shows that the violation chances are very rare, *e.g.*, even in the worst case less than 1% links have constraint violation after scheduling. These results show that in practice the scheduling algorithms work very well and make no or acceptable link constraint violations.

2) Different optimization goals: For the greedy and LP-based algorithms, we can choose to minimize the maximum link violation degree or to minimize the average link violation degree. Generally speaking, optimizing the worst case and the average violations may be conflicting with each other, however, we find that in our simulations the violation results (maximum and average link violation degree) are nearly the same, no matter which optimization goal is chosen. One possible reason is that the violations are very rare, and hence the two goals are nearly equivalent.

3) Varying link bandwidth constraints: Figure 11 shows the maximum link violation degree of the three scheduling algorithms under different link bandwidth constraints. We fix the number of rounds to be four. Clearly, when the link bandwidth constraints become tighter, the scheduling algorithm tends to have more violations. This is reasonable as the scheduling problem becomes harder when the resources are more limited. Figure 11 also shows even when the link constraints are set to be unreasonably small, the maximum link violations of the three algorithms are still acceptable.

3) Summary: The multi-round monitoring can significantly reduce the number of monitors, e.g., saving half for the four round scenarios. Also, the LP scheduling algorithm is able to schedule the path measurements with few violations even under extreme constraints.

4. Computation Speed Results

In this section we present the speed for monitor selection phase. The experiments described above were conducted on a machine with Intel(R) Xeon(TM) 2.80GHz CPU. For small VB topology, LP+Greedy costs about 10 hours to choose the monitors, while the greedy algorithm needs about 5 minutes to finish this process. For the other three large topologies, the monitor selection phase costs about 4 hours using greedy algorithm for single round. And for the scheduling problem, LP based algorithm needs most time, *e.g.*, half an hour, while the simple random and greedy algorithms need only several seconds and minutes, respectively.

6. Related Work

Generally, the experimental design of monitoring systems can be classified in two categories: path selection and monitor placement. In path selection approaches [4, 5, 7, 9, 17], the goal usually is to select minimal (or fix) number of path to satisfy (or maximize) the monitoring effect. In these approaches, the monitor placement is not considered or is too simple, but they usually have much complicated path selection goal, compared to the monitor placement approaches. For example, in [4] all end hosts in the overlay network are monitors and they do not consider selecting a subset of end hosts as monitors. Meanwhile the path sets corresponding to the algebraic basis of the path matrix is selected in [4]. In [5], SVD of the path matrix is further used to reduce the selected path set. Song *et al.* [7] introduced the Bayesian experimental design into network measurement. Their problem is to choose the best set of paths to monitor in order to achieve the highest expected estimation accuracy given the fixed total number of monitored paths. The operational constraints make the monitor and path selection problem very challenging, even if the selection goal is the simplest one, *i.e.*, to cover all the links. And it will be our future work to study other path selection goals in [4, 5, 7], which are more challenging under operational constraints.

The most related experimental designs in the literature are those monitor placement approaches for tomography [1, 18]–[22]. Bejerano *et al.* attempted to solve a simpler case of our monitoring selection problem [1], determining the smallest set of monitors whose probes can cover all the links in the network. Many important constraints such as monitor, replier and link constraints are not considered although this problem is still proved to be NP-hard in [1]. In [20] and [19], robustness problem is further considered to tolerate the routing dynamics. Nguyen *et al.* first determine the subset of paths to selection and then reduce the monitor placement problem to the vertex-cover problem. Besides, there are also some passive monitoring systems which select monitors for optimized SNMP polling [23] or traffic sampling [24]–[26], which are related but dealing different problems. Compared to the previous works, our experimental design problem is unique because of the consideration of the operational constraints. By enforcing the operational constraints, VScope takes into account the monitor and routing ability and avoids interfering with the normal network traffic in the heterogeneous VPN infrastructure. Our monitor placement problem is more like a constraint satisfactory problem instead of a pure optimization problem.

The constrained monitor selection problem may seem similar to some existing research topics such as placement of web cache replicas [27] or intrusion detection monitors [28]. But these problems usually only have the monitor constraints (*e.g.*, the load that monitors can take), while our problem faces much more complex constraints such as link bandwidth constraints and existing solutions cannot be applied. We found the classic network (call) admission control problem [29] is somewhat related to our problem in terms of the link bandwidth constraints. But unlike our problem, the admission control problem does not involve any monitor selection optimization.

7. Conclusions

In this paper, we propose VScope for continuously monitoring and diagnosis of VPN system under various operational constraints. The operational constraints are critical to ensure that the monitoring system itself will not disturb the normal traffic, especially in the heterogenous VPN infrastructure. We proposed the novel multi-round monitoring scheme for the monitor selection problem under operational constraints. The multi-round monitor selection has two phases, single-round monitor selection phase with relaxed constraints and the scheduling phase. Evaluation based on data obtained from real VPN and IP networks managed by a large tier-1 ISP demonstrate the efficiency and effectiveness of VScope.

8. Acknowledgement

We would like to thank Kevin D'Souza, K. K. Ramakrishnan, and anonymous reviewers for their valuable comments. This work was supported by DOE CAREER award DE-FG02-05ER25692//A001, and a Motorola grant CNVR017341. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding sources.

References

[1] Y. Bejerano and R. Rastogi, "Robust monitoring of link delays and faults in IP networks," in *IEEE INFOCOM*, 2003.
[2] V. Padmanabhan, L. Qiu, and H. Wang, "Server-based inference of Internet link lossiness," in *IEEE INFOCOM*, 2003.
[3] N. Duffield, "Simple network performance tomography," in *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2003.
[4] Y. Chen, D. Bindel, H. Song, and R. H. Katz, "An algebraic approach to practical and scalable overlay network monitoring," in *ACM SIGCOMM*, 2004.
[5] D. B. Chua, E. D. Kolaczyk, and M. Crovella, "Efficient monitoring of end-to-end network properties," in *IEEE INFOCOM*, 2005.
[6] Y. Zhao, Y. Chen, and D. Bindel, "Towards unbiased end-to-end network diagnosis," in *ACM SIGCOMM*, 2006.
[7] H. Song, L. Qiu, and Y. Zhang, "Netquest: A flexible framework for lange-scale netork measurement," in *ACM SIGMETRICS*, June 2006.
[8] V. V. Vazirani, *Approximation Algorithms*, Springer-Verlag, 2001.
[9] C. Tang and P. McKinley, "On the cost-quality tradeoff in topology-aware overlay path probing," in *IEEE ICNP*, 2003.
[10] Y. Zhao, Z. Zhu, Y. Chen, D. Pei, and J. Wang, "Towards efficient large-scale vpn monitoring and diagnosis under operational constraints," Tech. Rep. NWU-EECS-08-06, Univ. of Northwestern, Jul. 2008, http://cs.northwestern.edu/∼yzhao/.
[11] E. Tsang, *Foundations of Constraint Satisfaction*, Academic Press, 1993.
[12] D. S. Hochbaum and A. Pathria, "Analysis of the greedy approach in problems of maximum k-coverage," *Naval Research Logistics*, vol. 45, 1998.
[13] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear Programming and Network Flows*, Wiley-Interscience (3rd edition), 2004.
[14] R. Motwani and P. Raghavan, *Randomized Algorithms*, Cambridge University Press, 1995.
[15] T. R. Jensen and B. Toft, *Graph coloring problems*, Wiley-Interscience, New York, 1995.
[16] A. Medina, I. Matta, and J. Byers, "On the origin of power laws in Internet topologies," in *ACM Computer Communication Review*, 2000.
[17] M. J. Coates, Y. Pointurier, and M. Rabbat, "Compressed network monitoring for ip and all-optical networks," in *ACM/Usenix IMC*, 2007.
[18] J.D. Horton and A. Lopez-Ortiz, "On the number of distributed measurement points for network tomography," in *ACM/Usenix IMC*, 2003.
[19] Y. Breitbart, F. Dragan, and H. Gobjuka, "Effective network monitoring," in *International Conference on Computer Communications and Networks (ICCCN)*, 2004.
[20] R. Kumar and J. Kaur, "Efficient beacon placement for network tomography," in *ACM/Usenix IMC*, 2004.
[21] H. Nguyen and P. Thiran, "Active measurement for multiple link failures diagnosis in IP networks," in *PAM Workshop*, 2004.
[22] G. R. Cantieni and et. al., "Reformulating the monitor placement problem: Optimal network-wide sampling," in *Conference on Information Sciences and Systems (CISS)*, 2006.
[23] Li Li, M. Thottan, Bin Yao, and Sanjoy Paul, "Distributed network monitoring with bounded link utilization in ip networks," in *IEEE INFOCOM*, 2003.
[24] G. R. Cantieni, E. Fleury, I. Guerin Lassous, H. Rivano, and M.-E. Voge, "Optimal positioning of active and passive monitoring devices," in *ACM conference on Emerging network experiment and technology*, 2005.
[25] G. R. Cantieni, G. Iannaccone, C. Barakat, and C. Diot, "Reformulating the monitor placement problem: Optimal network-wide sampling," in *ACM CoNeXT*, 2006.
[26] K. Suh, Y. Guo, J. Kurose, and D. Towsley, "Locating network monitors: Complexity, heuristics and coverage," in *IEEE INFOCOM*, 2005.
[27] Y. Chen, L. Qiu, W. Chen, L. Nguyen, and R. H. Katz, "Efficient and adaptive Web replication using content clustering," *IEEE Journal on Selected Areas in Communications (J-SAC)*, vol. 21, no. 6, 2003.
[28] S. Noel and S. Jajodia, "Attack graphs for sensor placement, alert prioritization, and attack response," in *Cyberspace Research Workshop*, 2007.
[29] N. Alon, S. Gutner, and Y. Azar, "Admission control to minimize rejections and online set cover with repetitions," in *Proc. of SPAA*, 2005.