Efficient Traffic Flow Measurement for ISP Networks

Qinghua Wu, Zhenyu Li, Jianhua Yang, Gaogang Xie, Kavé Salamatian

Efficient Traffic Flow Measurement for ISP Networks

Qinghua Wu*†, Zhenyu Li*, Jianhua Yang*, Gaogang Xie*, Kavé Salamatian‡
*Institute of Computing Technology, Chinese Academy of Sciences, China
†Graduate School of Chinese Academy of Sciences, Beijing, China
‡LISTIC-PolyTech, Université de Savoie, France
{wuqinghua, zyli, jhyang, xie}@ict.ac.cn, kave.salamatian@univ-savoie.fr

***Abstract*—Traffic flow measurement is of great importance to ISPs for various network engineering tasks. An interesting problem is that how to determine the minimum number of links by monitoring which one can obtain the traffic flows of the whole ISP network. Previous works view the problem as Vertex Cover problem. They suffer from high time complexity and redundant monitoring. Different from these works, we study the problem from the perspective of edges and propose two models. The first model, *Extended Edge Cover* model, can determine the minimum set of monitored links, which are 30% less than that of previous works. The second model, *shared-path* model, is more suitable when the monitoring resources are limited but one still wants to measure a large part of the networks. Using this method, one can measure 85% of the network by monitoring 5% of links. Finally, we evaluate the performance of the two models through extensive simulations. The experimental results show the effectiveness and robustness of the two models.**

I. Introduction

In ISP networks, traffic flow measurement or monitoring is of great importance to network engineering tasks, such as anomaly detection, application traffic identification. To this end, the traffic flows on any link in an ISP network should be monitored. We can monitor a fraction of links and infer traffic flows in all the others. The problem is to determine the minimum number of links that should be selected. In practice, there may be not enough resources to monitor the links determined above. Thus another problem is that how to determine the monitoring links under budget constraints to infer the traffic flows for as many links as possible. We use *edge* and *link* interchangeably throughout this paper.

One way to solve the problems is to place monitors on routers and collect traffic flows of these routers [1], [9], [7], [2]. Following this way, some links would be monitored twice, and some links are unnecessarily monitored, whose traffic flows can be inferred from that of other monitored links. In this paper we monitor the ISP networks from *the perspective of edge*, meaning that we directly monitor links, instead of routers, to determine a minimum set of links, by which all other links' can be inferred. Following this perspective, no redundant link would be monitored.

We prefer to monitor as few links as possible and infer others' traffic flows from the monitoring results of these links. As mentioned before, two subproblems should be solved. The first one, also called *Minimum Cost Coverage Problem*, is to find the minimum number of links that should be monitored, in order to measure the whole network. We model this subproblem as to find an *Extended Edge Cover* set in the ISP network graph. A greedy algorithm is proposed to solve this problem. The number of monitoring links determined by this algorithm is almost 30% less than that of other methods.

The second subproblem is called *Resource Constraint Maximize Coverage Problem*. Previous works hardly consider this problem. In the case of resource limitation, how can one infer the traffic flows for as many links as possible by just monitoring a small fraction of links, subject to the resource constraints. To this end, we propose a *shared-path* model and give greedy heuristics to solve it. This model determines 5% of links to cover about 85% of the network.

The rest of the paper is organized as follows. Section II describes related work in this area. In Section III, the *Extended Edge Cover* model is proposed, followed by the *shared-path* model. The two models are evaluated in Section IV. Finally, Section V concludes the paper.

II. Related Work

Our work is closely related to traffic matrix estimation and traffic flow monitoring on routers.

A. Traffic Matrix Estimation

Traffic matrices reflect the traffic volume between OD-pairs in a network. In [6], estimation techniques are evaluated in estimation error and sensitivity. Zhao *et al.* in [11] propose a data streaming algorithm to process traffic stream into digest. Zhang *et al.* in [10] develop a spatio-temporal compressive sensing framework for traffic matrix estimation. Different from these studies, we focus on how to determine monitoring links to infer traffic flows.

B. Monitoring traffic flows from routers

In [1], Breitbart *et al.* show that Weak Vertex Cover can be used to formulate the bandwidth utilization problem and propose an approximation algorithm for this optimization problem. Suh *et al.* [9] formulate the problem that how monitors are placed to cost-effectively monitor traffic flows. All the above works study traffic flow monitoring problem from the perspective of *vertices* (*i.e. routers*). They are time-inefficient and suffer from redundant monitoring.

III. Two Proposed Models

In this section, we propose two models to solve the two sub-problems mentioned above: Minimum Cost Coverage Problem and Resource Constraint Maximize Coverage Problem.

A. Extended Edge Cover Model

We study the problem of *Minimum Cost Coverage Problem*, and propose the Extended Edge Cover model. An ISP network is present as a graph $G(V,\,E)$, comprising a set V of vertices and a set E of links. Considering a vertex v, traffic flowing into v in a period is approximately the same as that out of v([1]), which is stated as below:

$$\sum_{e \in E(v)} f_{in}(v,e) = \sum_{e \in E(v)} f_{out}(v,e) \quad . \tag{1}$$

where $E(v)$ represents the set of all edges adjacent to vertex v, $f_{in}(v,e)$ and $f_{out}(v,e)$ are flows on edge e, flowing into and out of v, respectively. The observation in Eq. 1 is called *flow conservation law*.

According to Eq. 1, for any vertex v obeying the flow conservation law, if it has only one edge e which is not monitored, then its traffic flows could be *inferred* easily by other edges adjacent to v, using Eq. 1. The inferred one can be further considered as monitored at the other end of edge. Our goal is to determine a minimum set of monitoring edges such that using the edges in this set we can infer the traffic flows of all other edges.

We say graph G is *closed*, if every vertex v in G obeys the flow conservation law. We define an edge as a *covered* edge if it is monitored or its traffic flows can be inferred from other monitored edges'. Let $deg(v)$ denote the degree of vertex v. *Extended Edge Cover* set is defined as follows.

Definition 1 (Extended Edge Cover set)**.** For a graph $G(V,E)$, an Extended Edge Cover set consists of the minimum number of edges that should be monitored in order to infer flows for all edges, by performing the following step iteratively:

find a vertex v with $(deg(v)-1)$ covered edges, set the last edge as covered.

Thus, the minimum set of monitoring edges can be determined by finding an EEC set in the network graph. The following theorem gives the size of such a set.

Theorem 1 (Extended Edge Cover)**.** For a closed graph $G(V,E)$, the size of Extended Edge Cover set is $|E|-|V|+1$.

Proof: Due to the space limit, we only provide the sketch of the proof. First, we show that an EEC set can be constructed using $|E|-|V|+1$ edges. Each edge of G is visited in arbitrary order. If the two nodes of a newly visited edge e can be connected by other visited edges, e is put into the EEC set. The number of edges in the EEC set is $|E|-|V|+1$.

Next, we prove that $|E|-|V|+1$ is the minimum. Suppose we have $|E|-|V|$ edges in the EEC set, then there are $2|V|$ edges (each unique edge is counted twice by the two vertices adjacent to it) to be inferred. We infer one edge each time. In any cases, there will be more than 1 vertex, each with 2 edges to be inferred. These edges could not be inferred any more. Thus, $|E|-|V|+1$ is the minimum. ■

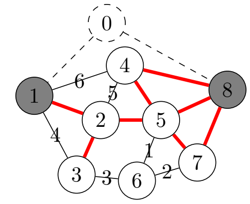

Fig. 1. Vertices in gray (*i.e.* vertex 1 and 8) do not follow Eq. 1. We add a dummy vertex 0 in dashed circle to make the graph closed. Red edges form the EEC set. Numbers on the black edges indicate the sequence of inferring process.

Theorem 1 guarantees that in a closed graph the size of Extended Edge Cover set is always $|E|-|V|+1$. In practice, the network graphs are not closed, since some vertices do not obey Eq. 1, such as ingress and egress routers in ISP networks. In such cases, we add one dummy vertex and connect it to vertices not following Eq. 1 by virtual edges with certain amount of flows. The new graph is closed. The number of monitored edges increases to $|E|-|V|+|OV|$, where OV is the set of vertices not obeying the conservation law. An example is shown in Fig. 1.

A greedy algorithm is proposed, in which edges are iterated. If a newly visited edge can be connected by other visited edges, the new edge is added into EEC set. There are $O(|E|)$ iterations in the algorithm. In each iteration, it takes $O(\log|E|)$ steps to check whether two nodes are connected by edges in C, using *union-find* set. The time complexity of the algorithm is $O(|E|\log|E|)$.

B. Shared-path Model

We propose a *shared-path* model to solve the *Resource Constraint Maximize Coverage Problem*, utilizing the shortest path property of the ISP networks.

In ISP network, from the perspective of shortest path routing protocol like OSPF and IS-IS, flows from origin s to destination t traverse the shortest path from s to t. Edges along a shortest path share the same flows, which means if the traffic flows of one edge in this path are monitored, traffic flows of other edges in this path can be inferred directly. An edge's flows are the sum of flows in all shortest paths traversing this edge. The problem is how to find a set of edges, such that by monitoring those edges, a very large fraction of other edges' can be inferred.

Given a graph $G(V,E)$ with p_i as shortest paths, and e_j as edges. Let $m_{i,j}$ be a $\{0,1\}$ indicator whether path p_i traverses edge e_j, $y_{i,j}$ represent whether path p_i is covered by edge e_j. Let x_j denote whether e_j is selected, c_j be cost for each edge, and B denote the resource constraint. The Resource Constraint Maximize Coverage Problem is NP-hard, which can be reduced from the budgeted maximum coverage problem [5].

Greedy heuristics are proposed. We define M as the set of already selected edges, and define edge's utility function $f(e)$

as the number of shortest paths covered by e but not covered by edges in M over the cost of e. The edges are iterated, until the resource limit is reached. Each time, we select the edge that maximize the utility function, and put it into M. The approximation factor of the algorithm, defined as the solution given by this algorithm over the optimal one, is bounded by Theorem 2. The proof is omitted due to the space limit.

Theorem 2. The greedy algorithm approximates the optimal solution within constant factor $1 - 1/e$.

There are $|E|$ iterations in the algorithm, where $|E|$ is the number of edges. In each iteration, it takes $O(|E|)$ time to find the edge that satisfies the condition. The time complexity of the approximation algorithm is $O(|E|^2)$. We will show that the algorithm performs rather well in the experiments.

IV. Experiment

In this section, we evaluate the Extended Edge Cover (EEC) model and shared-path model, showing the effectiveness and robustness of the models.

A. Dataset

TABLE I
ISP topologies provided by Rocketfuel

AS #	Name	#Routers	#Links
1221	Telstra (Australia)	318	763
1239	Sprintlink (US)	604	2279
1755	Ebone (Europe)	172	382
2914	Verio (US)	960	2828
3257	Tiscali (Europe)	240	404
3356	Level3 (US)	624	5301
3967	Exodus (US)	201	434
4755	VSNL (India)	11	12
6461	Abovenet (US)	182	296
7018	AT&T (US)	631	2078

We use the dataset of Rocketfuel to evaluate our models. Spring *et al.* [8] collect maps from ten various sized ISPs, and show the completeness of them. The data is shown in Table I.

B. Evaluating EEC Model

WVC [1] approximates the optimal solution within a factor of $1+\log|E|$ by reducing the rank of graph's matrix representation greedily. The time complexity of WVC is of the order of $O(|V|^2|E|)$. In contrast, our method is deterministic polynomial, with time complexity of the order of $O(|E|\log|E|)$, which is much lower than that of WVC.

The set of monitored edges is determined for each ISP network in Table I, using both WVC and EEC methods. The time each method consumes for ISP networks is shown in Table II. It shows that EEC uses far less time than WVC. In the experiment, WVC takes days of calculation to determine the monitored edges for large ISP networks (*e.g.* with more than 1,000 links). When the network scale is small, both methods consume nearly 0 sec. As the ISP networks grow larger, the time WVC consumes grows rapidly, while EEC takes only few seconds. When the networks have more than 1,000 links, WVC requires to use unacceptably long time.

TABLE II
Time comparison between EEC and WVC

AS #	#Edges	EEC	WVC
4755	12	0.03s	0.02s
6461	296	0.30s	5.0min
1755	382	0.34s	9.3min
3257	404	0.41s	15.5min
3967	434	0.40s	15.4min
1221	763	0.71s	42.4min
7018	2078	1.70s	>1day
1239	2279	1.81s	>1day
2914	2828	2.40s	>1day
3356	5301	3.82s	>1day

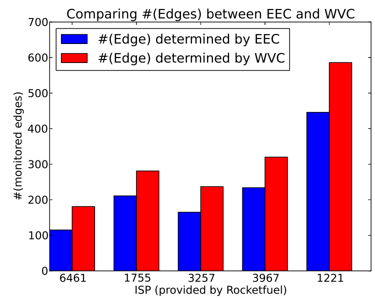

Fig. 2. Comparing number of monitored edges between EEC and WVC

In Fig. 2, we plot the number of monitored edges determined by EEC and that by WVC for 5 ISP networks. The ISPs are sorted increasingly according to the number of edges. Results for other ISP networks are not shown because WVC cannot calculate the edge set in reasonable time. Both the edge set determined by EEC and WVC are proportional to the scale of the network. Compared with WVC method, EEC uses about 30% fewer edges. In Fig. 2, ISP 3257 uses fewer monitored edges than ISP 1755. This is due to that the size of monitored edge set returned by the two methods is determined by both the number of vertices and that of edges in the network graph.

C. *Evaluating* Shared-path *Model*

First, the coverage ratio of the greedy heuristics is evaluated. In Fig. 3, we plot the fraction of monitored shortest paths versus the fraction of monitored edges for an ISP, using three methods: the greedy heuristics, selection by edge load and random selection. Results for other ISPs are the same as this one. The fraction of covered shortest paths determined by the greedy heuristics and edge load selection is much higher than that by random selection, using the same number of edges.

We define the number of additional shortest paths covered when a new edge is added as *marginal gain*. Marginal gain denotes the slope of the curves in the plot. From Fig. 3, when selecting few edges, both the marginal gain of greedy heuristics and that of selection by edge load are very high. Interestingly, greedy heuristics performs better than selection by edge load. This is because later added edges with high load may be present in many paths which have been covered by the already selected edges. When selecting about 5% edges, the greedy heuristics can cover almost 85% of the shortest paths.

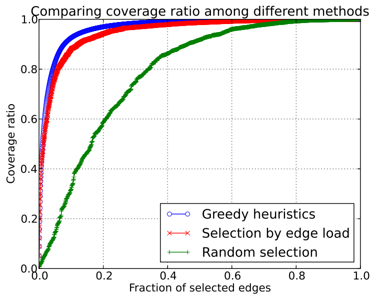

Fig. 3. Comparison of coverage ratio for different methods of selection

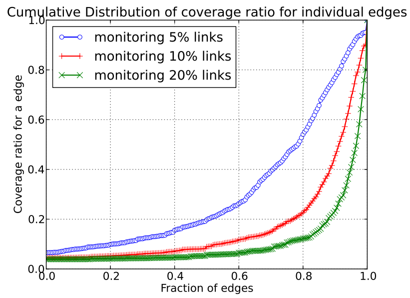

Fig. 4. Shortest path coverage ratio for individual edges

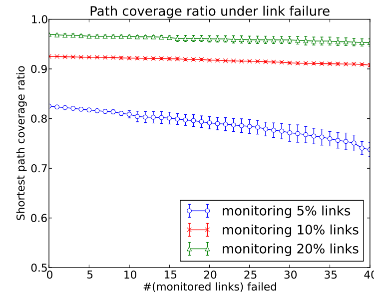

Fig. 5. Coverage ratio of shortest path under link failure

Then, we evaluate what percentage of traffic flows of an edge can be inferred. Greedy selection determines a small fraction of edges to cover a large fraction of shortest paths. However, an edge may be present in many paths. If we want to infer all the traffic flows of an edge, we must cover all the shortest paths it belongs to. When monitoring resource is limited, we need to calculate how many traffic flows of an edge can be inferred. We use the percentage of covered shortest paths for each edge (*i.e.* the coverage ratio for an individual edge) as the measure.

In Fig. 4, we plot the cumulative distribution that what percentage of shortest paths for individual edges can be covered when monitoring 5%, 10%, 20% links. As the line is closer to the lower right corner, more links have larger coverage ratio. From the figure, when monitoring more than 10% links, most links will be covered for more than 80% shortest paths. Even when monitoring 5% links, it can cover 75% shortest paths for more than 60% links. This shows great effectiveness of the shared-path model.

Next, we evaluate the shared-path model under link failure. Link failure is assumed to occur randomly with equal probability [3]. The shortest paths will change due to the update of routing tables, then the covered paths for each monitored link will change correspondingly. We only consider the failure of monitored links. We set 40 failures as the maximum, which are adequate to simulate the link failures in ISP network [4]. The experiments are run for 5 times, then we get the mean of coverage ratio as well as the standard error.

In Fig. 5, we plot the coverage ratio of all shortest paths when monitored links fail, monitoring 5%, 10%, 20% links. It shows that when monitoring 10% or 20% of links, the coverage ratio changes slightly under link failure. The remarkable result is that when monitoring 5% of links, the coverage ratio falls from 82.5% to 73.7% due to link failure. Thus, the coverage ratio is almost not affected by link failure. When a monitored link fails, the shortest paths it covers may be also covered by other monitoring links. This shows that the shared-path model is robust to link failure.

V. Conclusion

In this paper, we have proposed two models to measure the traffic flows in ISP networks. The Extended Edge Cover model, utilizes flow conservation for each router in network. By monitoring these links determined by EEC, the traffic flows of all other links in the network can be inferred accurately. The shared-path model, utilizes routing information of the network. It is suitable when the monitoring resources are limited. The two models have been evaluated through simulations with real-life ISP networks. The results have demonstrated the effectiveness and efficiency of the proposed models.

Acknowledgement

This work was supported by National Basic Research Program of China with Grant 2012CB315801, by National Natural Science Foundation of China (NSFC) with Grants 61133015 and 60903208, by Strategic Priority Research Program of the Chinese Academy of Sciences with Grant XDA06010303.

References

[1] Y. Breitbart, C. Chan, M. Garofalakis, R. Rastogi, and A. Silberschatz, "Efficiently monitoring bandwidth and latency in ip networks," in *INFOCOM 2001. Proceedings. IEEE*, vol. 2. IEEE, 2001, pp. 933–942.

[2] Y. Breitbart, F. Dragan, and H. Gobjuka, "Effective monitor placement in internet networks," *Journal of Networks*, vol. 4, no. 7, pp. 657–666, 2009.

[3] R. Durrett, *Random graph dynamics*. Cambridge Univ Pr, 2007, vol. 20.

[4] P. Francois, C. Filsfils, J. Evans, and O. Bonaventure, "Achieving sub-second igp convergence in large ip networks," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 3, pp. 35–44, 2005.

[5] S. Khullera, A. Mossb, and J. Naor, "The budgeted maximum coverage problem," *Information Processing Letters*, vol. 70, pp. 39–45, 1999.

[6] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot, "Traffic matrix estimation: Existing techniques and new directions," in *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4. ACM, 2002, pp. 161–174.

[7] V. Sekar, M. Reiter, W. Willinger, H. Zhang, R. Kompella, and D. Andersen, "Csamp: a system for network-wide flow monitoring," in *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*. USENIX Association, 2008, pp. 233–246.

[8] N. Spring, R. Mahajan, and D. Wetherall, "Measuring isp topologies with rocketfuel," in *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 4. ACM, 2002, pp. 133–145.

[9] K. Suh, Y. Guo, J. Kurose, and D. Towsley, "Locating network monitors: complexity, heuristics, and coverage," *Computer Communications*, vol. 29, no. 10, pp. 1564–1577, 2006.

[10] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4, pp. 267–278, 2009.

[11] Q. Zhao, A. Kumar, J. Wang, and J. Xu, "Data streaming algorithms for accurate and efficient measurement of traffic and flow matrices," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 33, no. 1. ACM, 2005, pp. 350–361.