A Heuristic Approach to Assess the Traffic Matrix of an ISP exploiting Segment Routing Flexibility

(Invited Paper)

Antonio Cianfrani, Marco Polverini, Trupti Nalawade

Department of Information engineering, Electronics and Telecommunications (DIET)
University of Roma “Sapienza” - Via Eudossiana 18, 00184 Roma, Italy
Telephone: +39 0644585371, e-mail: `name.surname@uniroma1.it`

***Abstract*— The Ingress Egress Traffic Matrix (IE TM) assessment is a fundamental step of the network management for an ISP network, since it represents the key input parameter used by any Traffic Engineering solution to optimize the resource utilization and to improve the Quality of Service. The actual TM assessment procedures are based on estimation algorithms or measurement based approaches. This paper presents a method to measure the intensity of traffic flows, that overcomes the limits of the classical measurement/estimation based approaches. The idea is to exploit the flexibility of the Segment Routing paradigm to implement controlled routing changes so that to measure the intensity of a subset of network flows. The main contribution of the work is to show the feasibility of the proposed approach by means of a low complexity heuristic, referred to as Path Cost Bases (PaCoB), able to identify the list of routing changes that allow to improve the TM assessment procedure. The heuristic is composed of successive steps, referred to as snapshots: in each snapshot the routing of a set of flows is changed so that to assess their intensities. The performance evaluation show that PaCoB assesses the intensity of more than 90% of flows. Moreover, when used in conjunction with an estimation algorithm, PaCoB allows to reduce the estimation error by more than 50% performing only 10 snapshots.**

***Index Terms*— Traffic Matrix, Traffic Measurement, Segment Routing**

I. INTRODUCTION

The knowledge of the traffic demand of an Internet Service Provider is a fundamental step for the implementation of advanced Traffic Engineering solutions able to improve the Quality of Service (QoS) offered to end customers. The traffic demand can be represented by means of a traffic relationship matrix, referred to as the Ingress-Egress Traffic Matrix (IE TM, simply TM in the following): each element of the TM represents the amount of traffic crossing the network from a specific Ingress router to a specific Egress one, i.e. a IE flow.

The TM assessment problem is a well known problem for the research community but it is still an open issue [1]. The classical approaches are the ones based on estimation algorithms [2]: the traffic relationships are modeled using a mathematical function and the TM is estimated having as input parameter the link loads and the network paths. A different kind of solutions is represented by the measurment based approaches: exploiting a specific technique, such as SDN [3], [4] or Netflow [5], it is possible to obtain a greater amount of input, i.e. Origin Destination flow intensities, so that to reduce the number of unknown elements of the TM. Anyway, both solutions have drawbacks: the estimation algorithms are dependent on the effectiveness of the mathematical model used [6], while measurements-based solutions have scalability issues and are not able to identify IE flows.

In this way we propose a different approach to assess the TM of an ISP: introducing controlled routing changes so that to obtain load variations on network links and, consequently, to provide additional inputs for the TM assessment problem. The general idea is then translated into an heuristic, referred to as Path Cost Based (PaCoB) algorithm, able to identify an ordered set of snapshots: each snapshot is a list of routing changes, to be performed at the same time, able to identify the traffic intensity of a set of IE flows. PaCoB is able to improve the TM assessment procedure assuring the Quality of Service level, i.e. network congestion is avoided.

The feasibility of our approach is based on the availability of the novel Segment Routing (SR) [7] paradigm. SR allow to implement advanced Traffic Engineering (TE) features on top of a pure IPv6 network with no need of flow-based protocols, such as Multi Protocol Label Switching (MPLS). With respect to our proposal, SR allows to modify network paths in a easy and controllable way, acting only at ISP border routers without impacting the network performance.

The routing modification idea was already proposed by Nucci et al. in [8], where an algorithm to modify Interior Gateway Protocol (IGP) links weights so that to trigger new paths and consequent links load variations was defined. The proposed solution, referred to as IGPWE (IGP Weight hEuristic), is focused on the link weigth assignment and is constrained by the distributed nature of path computation in an classical IGP scenario. Our solution is defined in a source-based routingscenario, i.e. SRv6, and the its aim is to detect new paths, not new link weigths. Moreover, thanks to the exploitation of SR, PaCoB is also able to overcome the implementation drawbacks of the algorithm in [8], mainly due to the convergence phase caused by link weights modification.

To summarize, the main contributions of the paper are:

- the proposal of a new approach for the TM assessment based, based on controlled routing changes;
- the definition of a fast heuristic able to identify the set

978-0-9883045-5-0/18/$31.00 ©2018 ITC
DOI 10.1109/ITC30.2018.00037

of paths to be changed so that to assess an high number of IE flows;
- a performance evaluation showing the effectiveness of our proposal.

The rest of the paper is organized as follows. In Section II an overview of the Segment Routing paradigm is provided, while in Section III the controlled routing modification idea is introduced. In Section IV PaCoB algorithm is presented and in Section V the performance evaluation is reported. Finally, in Section VI conclusions are provided.

II. THE SEGMENT ROUTING PARADIGM

The Segment Routing (SR) [7] paradigm has been proposed in the last years to improve the Traffic Engineering capabilities of an IP network. The aim of SR is to provide a mechanism for the definition of advanced features, such as TE paths and Network Function Virtualization (NFV) support, without using complex flow based protocols, such as Multi Protocol Label Switching (MPLS). SR can be implemented over an MPLS or an IPv6 network. In the following we consider only the last case (i.e. SRv6), even if the same concepts can be applied also for an MPLS network.

The main features of SR that we exploit in the paper to propose our TM assessment solution are: i) the explicit routing and ii) the source routing. To better explain these aspects, we need to introduce the relationship among SR paths and IPv6 paths. The IPv6 paths, computed using a classical Interior Gateway Protocol (IGP), are used by SR to ”construct” SR paths, i.e. the paths used by the traffic as a result of a specific TE optimization. Thus, an SR paths is defined using a Segment List, i.e. a list of nodes to be crossed from the incoming router to the outgoing edge: the path among two consecutive routers in the Segment List is the IPv6 paths.

The routers of a SRv6 domain are identified by the so-called Segment IDentifier (SID), i.e. one of the router’s IPv6 addresses. In this way, it is possible to code any network path using a Segment List (explicit routing): in the simplest case, the whole set of routers crossed can be present in the Segment List; otherwise, exploiting the available IPv6 paths, only a subset of routers can be inserted [9].

Any packet entering an SR domain is encapsulated into an SR header by the incoming edge router. In the SR header, a field is dedicated to the Segment List; the network routers will be able to read the Segment List and to forward the packets on the basis of the configured SR path. In this way, considering all the traffic entering an SR network through a specific incoming edge router, the packet classification and consequent path definition (by means of the Segment List definition) are executed only by incoming edge router. Thus, it is possible to modify the path for a specific IE flow simply modifying the Segment List at the incoming edge. This feature of SR is the key point for the feasibility of our TM assessment solution.

III. CONTROLLED ROUTING PERTURBATION FOR TM ASSESSMENT

The Ingress Egress Traffic Matrix (TM) assessment is a long standing problem for the network management research community. In this Section we first provide a brief overview of actual techniques for TM assessment; then, we describe the idea modifying SR paths to assess IE flows; finally, we highlight the difference among our solution and a similar one, proposed for a pure IP network.

A. TM assessment overview

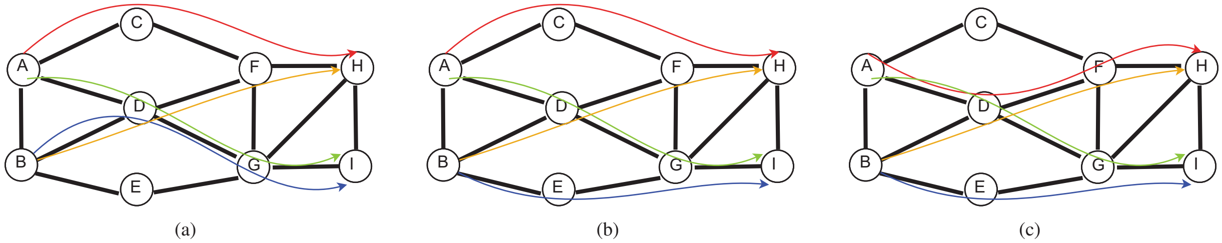
The main approaches proposed in the past for TM assessment can be classified in two categories: i) estimation-based algorithms, and ii) measurements based solutions.

The estimation based algorithms are able to use as input only the links loads information: the estimated TM is obtained applying a mathematical model to characterize the IE traffic relationships. In this work we consider the well-known tomogravity estimation algorithm [2]: it first computes a rough estimation of the TM by adopting a gravity model (the intensity of a flow is proportional to the product among the amount of traffic entering or leaving the network by means of the end points of the considered flow) and then it selects the TM that both minimizes the least square distance with respect to previous TM (obtained using the gravity model) and is coherent with respect to the routing and the current links loads. The main weak point of all estimation based algorithms is that the final results is strictly correlated to the traffic model considered and the final estimation error could be significant.

On the other side, measurement based solutions exploit the availability of specific Origin-Destination (OD) flow counters in network nodes so that to obtain traffic information at an higher granularity level with respect to link loads. Two main measurements based solutions have been proposed: Software Defined Networks (SDN) and Netflow ones. In SDN, the switches provide a counter associated to each forwarding table row and so it is possible to insert ”measurement rows”, i.e. rows only used to provide input for the TM assessment procedure; the measurement rows must be few, since the forwarding table of switches ha a limited size, and different selection criteria have been proposed [3], [4]. IPFIX and Netflow [10] are monitoring protocols able to extract at router level statistics about OD flows and to collect all the measurements at a central node for a final computation. In both cases, the measurements are related to OD flows, and not to IE flows: thus, extracting the IE flow level information from OD flows requires cooperation with the routing module, i.e. BGP routing table queries.

B. Traffic changes and SR for TM assessment

Considering all the previous issues, we propose in this paper a totally different approach to assess the TM of an ISP. The idea is to change network paths in a controlled way so that to obtain new information from link loads. In particular, when a new routing scheme is implemented, the IE flows will change their paths and the final outcome will be new loads on the network links. In the simplest case, if the path of a single IE flow (simply flow in the following) is changed then the load of a subset of network links is changed: the links of the old path will experience a load decrease equal to the flow demand, while the links of the new path will experience a load increase



(a)

(b)

(c)

Fig. 1. Example of path modification for a simple network composed of 8 nodes and 14 links.

equal to the flow demand. It is clear that considering only one flow involved by the routing change, it is possible to easily assess the flow demand.

In Fig. 1, a simple network is shown to explain the path modification idea. The network considered is composed of 9 nodes (from A to I) and 14 links: for the sake of simplicity, only four IE flows are shown, i.e. f_{AH} (red line), f_{AI} (green line), f_{BH} (orange line) and f_{BI} (blue line). In Fig. 1(a), the initial routing configuration for the considered flows is reported. In Fig. 1(b), a path modification involving f_{BI} is performed: the new path of f_{BI} is the one crossing nodes E, G and H. When passing from the routing configuration of Fig. 1(a) to the one of Fig. 1(b), it is possible to assess the demand of f_{BI} evaluating the load increase on link BE (or on link EG). The SRv6 protocol allows to implement the path modification acting on the Segment List (SL) attached to traffic from node B to node I: the (simplest) new SL to apply in B, using the node letters as SIDs, is $[E, G, I]$, where each crossed router is reported in the list [1]. In Fig. 1(c) a new path modification is shown: flow f_{AH} is moved to a new path (encoded by the SL $[D, F, H]$ in A), and its demand can be assessed evaluating the load decrease on link AC (or on link CF).

Starting from this idea, we define here a way to assess the whole TM by performing controlled network paths modification. With respect to the (simple) single flow modification case, we try to answer to the following questions: is it possible to modify the path of a set of flows at the same time? How to differentiate the contribution of the different flows on the new link loads? How many paths modifications are needed to obtain the full TM? And finally, is it possible to integrate the path modification solution with a simple estimation algorithm to assess the TM in a more practical way?

Before answering the previous questions, we highlight a very important aspect: the flexibility provided by the Segment Routing to implement our solution.

As explained in Section II, one of the main feature of SR is to provide a source routing: it is possible to specify the network path simply modifying the Segment List "attached" to incoming packets at the edge of the network. This also means that a controlled routing modification regarding a specific flow is obtained with a single operation: the incoming router will have a new Segment List associated to the flow. There will be no need of "involving" the other network nodes and the modification will be available immediately. In this way, SR represents the ideal scenario where to implement the controlled routing modification mechanism for TM assessment.

C. IGPWE

The idea of modifying network paths to obtain extra-information from link loads has already been proposed by Nucci et al. in 2004. The network scenario considered is the one of an IP network where Open Shortest Path First (OSPF) is used as intra-domain routing protocol. The controlled path modifications are realized modifying the OSPF link weights. The authors define an iterative heuristic, named IGPWE (IGP Weight hEuristic): at each step a set of weights is modified so that to use new network paths and obtain new link loads. The results reported shows that after a certain amount of steps, referred to as snapshots, it is possible to assess the intensities of a high number of network flows.

The IGPWE heuristic is different than the one proposed here. As first, the problem of choosing the OSPF weights in a pure IP network is constrained by the distributed nature of path computation: each link weigth modification impacts all the paths using such a link; in our case, the flexibility of a source based routing solution, allows to independently compute IE paths. Moreover, the IGPWE has a main drawback: modifying the OSPF weights leads to a network convergence state [11]. During the convergence, each network router must i) receive the new message about weights change, ii) execute the Dijkstra algorithm from scratch (to recompute network paths), and iii) update the routing table. Considering that the whole procedure is distributed and the routers are not synchronized, during the convergence unexpected network paths can arise leading to performance degradation.

IV. IMPLEMENTING THE PATH MODIFICATION IDEA

Before describing the algorithm proposed to implement the controlled routing change idea, let us introduce few notations. The SR network can be represented by means of graph $G(V, L)$, where V is the set of N nodes and V is the set of E links; the generic link l is characterized by a capacity C_l. We focus our attention on IE flows, referred to as f_i ($i = 1, ..., N(N-1)$); in the general case, the number of IE flows is equal to $N(N-1)$, i.e. there is a traffic relationship among all nodes pairs. Each flow f_i is routed using a network path $p(f_i)$, computed by a Traffic Engineering algorithm to optimize a specific objective function. The input of a generic

---

[1] shorter SLs can be obtained considering the available IP shortest paths, but this is out of scope here.

TM assessment problem is represented by the network path $p(f_i)$ $(i = 1, ..., N(N-1))$ and the link loads $\rho(l_h)$ $(h = 1, .., E)$. When a new path $p'(f_i)$ is "forced" for flow f_i, new link loads $\rho(l^*)$ are obtained for the links of the old path (i.e. $l^* \in p(f_i)$) and for the links of the new path (i.e. $l^* \in p'(f_i)$).

A. Compatibility and Bandwidth Constraints

As already stated in the previous Section, when several path modifications (i.e. involving several flows) are executed at the same time, it is important to define a way to differentiate the different flows contributions to the new links loads. In this work we define the following constraint to allow for multiple path modifications, referred to as **compatibility constraint**: for each "modified" flow f_i, it must be possible to identify a link, referred to as *monitoring link* for flow f_i, where f_i is the unique flow crossing the link that have experienced a path modification. The monitoring link is univocally associated to the flow and it will be used to perform the flow measurement. In other words, if a set of flows is involved by path modifications, there must exist at least one monitoring link for each considered flow.

A further constraint to be considered when performing a path modification is the **bandwidth constraint**: when moving a flow f_i from path $p(f_i)$ to path $p'(f_i)$, it must be assured that there is enough available bandwidth for f_i on $p'(f_i)$. It is clear the the bandwidth constraint must be assured to avoid network congestion; anyway it is not easy to implement it, since the intensity of f_i is not known (it will be the output of the heuristic). In our solution we define a worst-case rule to assure the bandwidth constraint: the flow intensity is over-estimated considering the load of the link having the lower bandwidth usage on the old path, i.e. $D(f_i) = min\{\rho(l_j) * C_{l_j}\}$ where $l_j \in p(f_i)$.

The algorithm we defined will respect both compatibility and bandwidth constraints. The algorithm functioning is based on the snapshot concept (as for [8]): a snapshot is a set of path modifications to be performed at the same time, that allow to measure a specific set of flows. The outcome of the algorithm is an ordered set of snapshots: at each snapshot, i) a set of flows is selected, b) a routing modification is performed for each selected flow, and iii) the new link loads on the monitoring links are evaluated to assess the flows intensity.

B. PaCoB Algorithm

The Path Cost Based (PaCoB) heuristic is a low complexity iterative algorithm able to detect the set of flows, along with alternative paths, to be involved in the controlled routing modification procedure. The PaCoB algorithm is composed of four phases:

- the computation of alternative paths;
- the ranking of flows;
- the flows and monitoring links selection;
- the flows measurements.

The *computation of alternative paths* is performed offline: the well-known K-Shortest Path algorithm is executed for each flow, and consequently a list of k alternative paths is available for each flow.

Algorithm 1 Pseudocode of the third phase of PaCoB algorithm for the h-th snapshot.

```
Input: $FPS_{h-1}$
$FPS_h = FPS_{h-1}$
$L_h = \emptyset$
$F_h = \emptyset$
for $j$ : 1 to size of ($FPS_h$) do
    extract next item $(f_i, p_k, s_k)$ from $FPS_h$
    if ($\exists l_m \in p_k$ so that $l_m \notin L_h$) AND (bw_chk($f_i$) = true) then
        add $l_m$ to $L_h$
        add $f_i$ to $F_h$
    remove items $(f_i, p_k, s_k)$ with $k = 1, .., K$ from $FPS_h$
Output: $F_h$, $L_h$, $FPS_h$
```

The *ranking of flows* is performed considering all the alternative paths associated to each flow. More in detail, each flow - alternative path pair has a score assigned. PaCoB will use as score an integer value representing the number of links not in common among the original path of the flow and the alternative path. The flows are ordered in ascending order of the score value, so that the flows having new paths with many links in common with the original one will be preferred during the algorithm execution: the idea is to try to minimize the impact of new paths on the network (since new paths are "similar" to original ones) and so to increase the probability of finding flows to be re-routed and related monitoring links. The flows ranking phase can be performed offline along with the alternative paths computation.

The third phase, i.e. *the flows and monitoring links selection* is the core of PaCoB heuristic. As already said, PaCoB provide as output an ordered set of snapshots, referred to using the notation SN_h with $h = 1, ..., M$ (M is the number of snapshots). The output of each snapshot SN_h is the set of flows to be measured F_h, with related alternative paths, and the set of monitoring links L_h (one for each flow). Let us focus on a generic snapshot SN_h, having as input the set FPS_{h-1} composed by the elements (f_i, p_k, s_k), where f_i is the i-th flow not measured yet, p_k is the k-th alternative paths available for f_i, and s_k is the score value associated to p_k. The pseudocode of the flows and monitoring links selection phase is reported in Algorithm 1.

The set FPS_h, representing the set of flows with alternative paths that have not been measured in the previous snapshots, is ordered on the basis of the score value. Thus, the first element (f_i, p_k, s_k) of the set is chosen, i.e. the one having the lowest score value, and the alternative path is considered for the flow. Then, a monitoring link is searched for flow f_i: if a link l_m of p_k satisfying the monitoring link constraint ($\exists l_m \in p_k$ so that $l_m \notin L_h$) and the bandwidth constraint (bw_chk(f_i) = true) is found, then f_i is inserted in the set F_h of flows measured at snapshot h-th, and the link l_m is inserted in the set L_h of monitoring links at snapshot h-th. Otherwise, the next flow of FPS_h is considered. At the end of the snapshot h-th, all the flows in F_h are removed from FPS_h, since there is no need of considering them during the next snapshots.

The last phase is *the flows measurements* one. The output of each snapshot, i.e. F_h and L_h, is used to perform routing modifications and link load measurements. This last phase

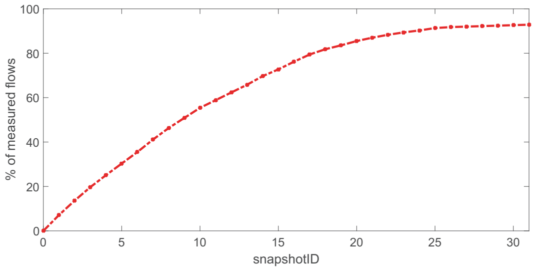


Fig. 2. Percentage of measured flows by PaCoB for the Geant network.

only requires to update the Segment Lists associated to the flows in F_h and to collect, using router telemetry (or even the SNMP protocol), the load of links in L_k. Following the snapshots order, a dedicated flows measurements procedure will be performed for each snapshot.

V. PERFORMANCE EVALUATION

In this section we provide the performance evaluation of the PaCoB algorithm. We implemented PaCoB on Matlab and considered two networks from the Survivable fixed telecommunication Network Design repository [12], named Nobel ($N = 17$, $E = 52$) and Geant ($N = 22$, $E = 72$). For those networks real traffic matrices are available. The initial routing is supposed to be selected according to a shortest path rule.

In order to set up links capacities, we first route the considered TM using the initial routing, then we assign to each link an available bandwidth equal to twice the amount of carried traffic. Finally, the set of alternative paths is built considering a value of $K = 5$ for the K-Shortest-Path (KSP) algorithm.

In the following we shows three different analysis to characterize the performance of PaCoB: i) the evaluation of the amount of flows assessed at each snapshot; ii) the impact on the network performance, in terms of path length increase; iii) the TM assessment performance of a solution integrating PaCoB with the tomogravity estimation algorithm [2].

In Fig. 2 the percentage of measured flows as a function of the number of snapshots performed for the Geant network is reported. Considering that the overall number of IE flows for Geant is equal to 462, the results obtained highlight that PaCoB is able to measure a significant amount of flows for each snapshot. In particular, after only 7 snapshots more than 50% of network flows are measured. A further consideration regards the maximum number of flows assessed: PaCoB is not able to measure all the flows (after 31 snapshots, the 92% of flows intensities are available). The reason for such an outcome is that the bandwidth checking constraints the number of available paths modification and so about the 5% of flows are not measurable.

A similar evaluation is reported for the Nobel network in Fig. 3. The results are, in general, similar to the ones obtained for Geant. The main difference is represented by the maximum number of measured flows: in the Nobel case, after 41 snapshots less than 70% of flows are measured. This means that the bandwidth constraint has an higher impact in the Nobel case with respect to Geant one. Anyway, even in this case, the 50% of network flows are known after few snapshots.

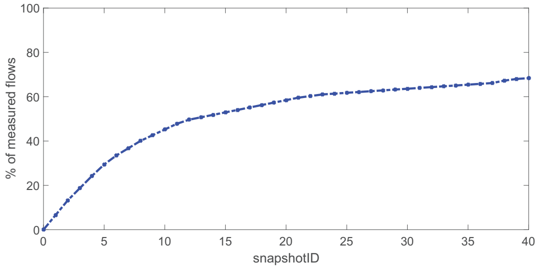


Fig. 3. Percentage of measured flows by PaCoB for the Nobel network.

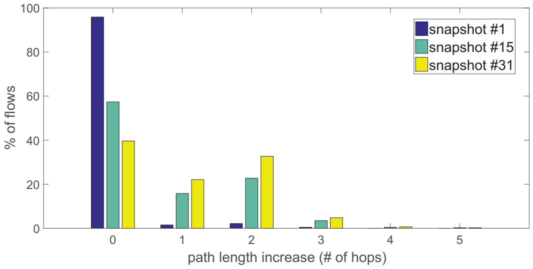


Fig. 4. Percentage of measured flows by PaCoB for the Nobel network.

The side effect of PaCoB is to move network flows from TE paths, that usually are the shortest ones, to sub-optimal ones. To evaluate the performance degradation introduced by our solution, we evaluated the path length increase, in terms of number of hops, when a certain amount of snapshots have been performed. In Fig. 2 the path length increase in the Geant network, for three different snapshot values, is reported. Fig. 2 shows that when only the first snapshot is performed, the 5% of flows experiences a path length increase. The percentage of flows having a longer path than the initial one is 45% and 60% after 15 and 31 snapshots, respectively. Anyway, the path length increase is greater than 3 hops for a negligible number of flows in all the considered cases.

The last analysis reported regards the possibility of combining PaCoB with the tomogravity estimation algorithm. The idea is to perform a pre-fixed number of snapshots and then, using the flows demand values measured, executing the tomogravity algorithm. The results obtained are reported in Fig. 5: for each snapshot value, the Relative Root Mean Squared Error (RRMSE) [13] of the estimated TM is computed. The same evaluation is reported for the IGPWE algorithm. The first consideration is that the integration of PaCoB with tomogravity provides a viable approach in a real scenario: performing a limited amount of snapshots, it is possible to greatly reduce the estimation error. In the Geant case, using the outcome provided by the first 10 snapshots it is possible to reduce the estimation error by 50%. Moreover, PaCoB shows better results with respect to IGPWE. The main reason is that IGPWE has a significant constraint: a link weight change affects many paths and it is not possible to act on single paths, due to the distributed path computation strategy. In PaCoB, SR

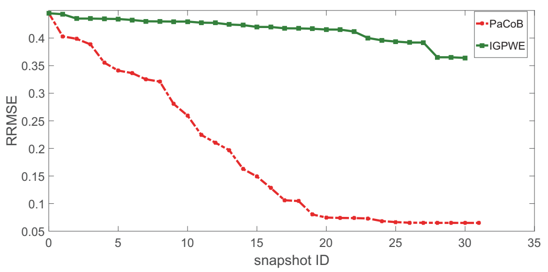


Fig. 5. Percentage of measured flows by PaCoB for the Nobel network.

is exploited to act only on specific flows, without impacting the routing of different ones.

VI. CONCLUSIONS

In this paper we have proposed a new way to improve the TM assessment procedure for an ISP network. Our idea is to exploit the Segment Routing flexibility to implement controlled path changes and so to collect new link loads. To show the feasibility of our solution, we defined a fast heuristic, referred to as PaCoB. PaCoB identifies an ordered set of path changes, organized into snapshots, able to provide a direct measurement of specific IE flows; PaCoB also assures that link congestion is avoided.

The performance evaluation shows that a significant number of flows is assessed by performing few snapshots. Moreover, combining PaCoB with the classical tomogravity estimation algorithm, it is possible to greatly reduce the error of the estimation procedure.

As future steps, we plan i) to formally define the problem of minimizing the number of snapshots to assess the TM, ii) to evaluate the impact of traffic variations on the performance of our solution, and iii) to deploy a real testbed for a deeper evaluation of the QoS degradation due to path modifications.

REFERENCES

[1] P. Tune and M. Roughan, "Internet Traffic Matrices: A Primer," in *Recent Advances in Networking, Volume 1*, H. Haddadi and O. Bonaventure, Eds. ACM SIGCOMM eBook, 2013.

[2] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast Accurate Computation of Large-scale IP Traffic Matrices from Link Loads," *SIGMETRICS Perform. Eval. Rev.*, vol. 31, no. 1, pp. 206–217, June 2003.

[3] M. Malboubi, L. Wang, C. nee Chuah, and P. Sharma, "Intelligent SDN based traffic (de)Aggregation and Measurement Paradigm (iSTAMP)," in *INFOCOM, 2014 Proceedings IEEE*, April 2014, pp. 934–942.

[4] M. Polverini, A. Iacovazzi, A. Cianfrani, A. Baiocchi, and M. Listanti, "Traffic matrix estimation enhanced by sdns nodes in real network topology," in *2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2015, pp. 300–305.

[5] C. Estan, K. Keys, D. Moore, and G. Varghese, "Building a Better NetFlow," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, pp. 245–256, Aug. 2004.

[6] S. Eum, J. Murphy, and R. Harris, "A Failure Analysis of the Tomogravity and EM Methods," in *TENCON 2005 2005 IEEE Region 10*, Nov 2005, pp. 1–6.

[7] C. Filsfils, S. Previdi, L. Ginsberg, B. Decraene, S. Litkowski, and R. Shakir, "Segment Routing Architecture," Internet Engineering Task Force, Request for Comments 8402.

[8] A. Nucci, R. Cruz, N. Taft, and C. Diot, "Design of igp link weight changes for estimation of traffic matrices," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 4. IEEE, 2004, pp. 2341–2351.

[9] A. Cianfrani, M. Listanti, and M. Polverini, "Translating Traffic Engineering outcome into Segment Routing paths: The Encoding problem," in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2016, pp. 245–250.

[10] R. Hofstede, P. Celeda, B. Trammell, I. Drago, R. Sadre, A. Sperotto, and A. Pras, "Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 4, pp. 2037–2064, Fourthquarter 2014.

[11] L. Chiaraviglio, A. Cianfrani, M. Listanti, L. Mignano, and M. Polverini, "Implementing energy-aware algorithms in backbone networks: A transient analysis," in *2015 IEEE International Conference on Communications (ICC)*, June 2015, pp. 142–148.

[12] (2014) SNDLib library. [Online]. Available: http://sndlib.zib.de/home.action

[13] A. Gunnar, M. Johansson, and T. Telkamp, "Traffic Matrix Estimation on a Large IP Backbone: A Comparison on Real Data," in *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '04. New York, NY, USA: ACM, 2004, pp. 149–160.