

An Integrative Multi-level Analysis of Internet Path Changes

Ang Chen[§], Edmond W. W. Chan^{†‡}, Xiapu Luo[§], and Rocky K. C. Chang[§]
 Department of Computing[§] Corporate Research Department[†]
 The Hong Kong Polytechnic University Huawei Research, China
 {csachen|csxluo|csrchang}@comp.polyu.edu.hk edmond.chan@huawei.com

ABSTRACT

Quantifying path changes from multiple granularity levels is often required. However, existing methods all characterize path changes on each level independently, thus duplicating and/or missing certain path differences. In this paper, we propose DiffP, an integrative multi-level analysis to overcome this deficiency, and demonstrate its applications in analyzing path changes during an ISP transition.

Categories and Subject Descriptors: C.2.3 [Computer-Communication Networks]: Network Operations

General Terms: Measurement, Performance

Keywords: Traceroute, Route change

1. INTRODUCTION

Identifying the scope of route changes from *multiple granularity levels* (e.g., subnet, AS) is an important requirement in Internet measurement. For example, Pathak et. al. [1] characterize the differences between two paths by assessing their AS-level and router-level discrepancy. In this paper, we discuss why segregative analysis on each granularity level, as employed by existing works, leads to information entanglement (§2). We then propose DiffP, an integrative analysis method based on a just-enough-resolution technique (§3). Given two snapshots of an end-to-end path and any set of target granularity levels, DiffP can accurately report the number of different entities on each level, and characterize a route-sharing structure as seen in an integrative view. We use it to tightly correlate path and performance changes in our ISP transition analysis (§4).

[†]This work was done when the author was with The Hong Kong Polytechnic University.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ACM CoNEXT Student Workshop, December 6, 2011, Tokyo, Japan.
 Copyright 2011 ACM 978-1-4503-1042-0/11/0012 ...\$10.00.

2. EXISTING SEGREGATIVE ANALYSIS

A multi-level analysis is often required for identifying the change scope between two network paths (e.g., [1]). However, the existing methods process the levels one by one, and thus are still restricted to a single level at a time. Since the single levels remain independent from each other, the differences identified at all levels cannot be integrated effectively. Figure 1(a) depicts the route-sharing structure of two paths from S to D on IP, subnet (SN), and AS levels. In contrast, Figure 1(b) compares them on each level separately, by calculating a longest common subsequence (i.e., LCS, presented as the “boxed” nodes). Moreover, nodes in the same entity (e.g., SN/AS) are treated as a whole. We cannot easily identify the route-sharing structure any more. Differences in one level (e.g., A on AS level) are duplicated into lower levels (e.g., C on SN and F on IP levels), while the similarities (e.g., B on AS level) miss lower level discrepancy (e.g., D on SN and H on IP levels).

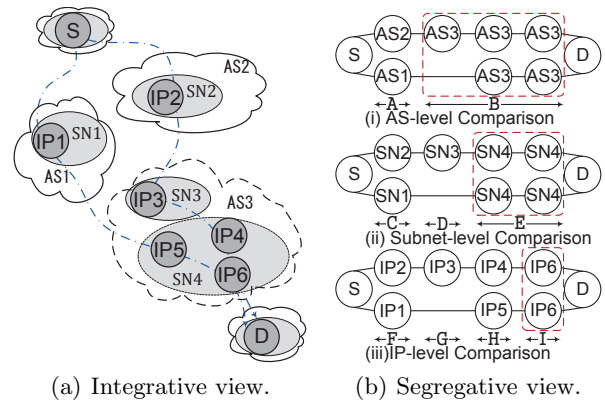


Figure 1: Differences between two paths using integrative and segregative comparisons.

Omissions and duplications of differences come from comparing with a *fixed resolution*. None of the fixed resolutions is the best for comparing the whole depicted paths. The AS-level is the best for comparing segment pair A, because lower levels are too granular to see that they actually do through distinct ASes. But D is accu-

rately characterized only by SN-level, because AS granularity is too coarse to capture any difference and IP too fine to see the traversals through different SNs. IP-level, on the other hand, best suits H and I, because no other resolution can reveal such low-level information.

3. A NEW INTEGRATIVE ANALYSIS

We propose `DiffP` (Algorithm 1), an integrative analysis of path changes with *just enough resolutions*. It starts with the lowest resolution and computes the LCS of two paths on AS level. If this resolution fails to identify any difference in a segment pair (i.e., in LCS portion such as B), it will switch to a finer resolution (i.e., SN level) for that pair to capture possible difference there. The resolution switch is implemented by recursively comparing current-level LCS segment pairs on the next finer level, also using LCS-based comparison. On the other hand, if the current resolution already identifies the difference (i.e., in non-LCS segment pairs such as A), `DiffP` will not tune up the resolution, because doing so will duplicate the differences into the lower level. Finally, the recursion returns when it finishes comparing with the finest granularity (i.e., IP level in our case). In this way, each segment pair is compared using the fittest resolution.

Algorithm 1 `DiffP(rt_pair, l)`; $l \in \{AS, SN, IP\}$; initially $l=AS$; `rt_pair` is the two paths to compare.

```

1: (lcs_pairs, nonlcs_pairs)  $\leftarrow$  LCS(rt_pair,  $l$ );
2: diff_cnt[ $l$ ] +=  $|\text{nonlcs\_pairs}|$ ;
3: if  $l == IP$  then
4:   return
5: end if
6: next_l  $\leftarrow$  Get_Next_Level( $l$ );
7: for all lcs_pair  $\in$  lcs_pairs do
8:   DiffP(lcs_pair, next_l);
9: end for

```

`DiffP` outputs the number of different entities on each level: for the example paths there are two distinct ASes (i.e., AS1, AS2), one distinct subnet (i.e., SN3), and two distinct IPs (i.e., IP4, IP5). We further quantify the overall difference by weighing each difference level, summing all weighted differences, and normalizing by the product of path lengths and the highest weight. The score increases with the path differences: zero for two identical paths and one for two paths without a common AS. Currently we set the weights of AS:SN:IP to be 9:7:2, but other weighting is also possible. Moreover, `DiffP` simultaneously characterizes the integrative route-sharing structure and reports the location of the first IP-level divergent node, which could be used to approximate the change origin in PlanetSeer [3] system.

4. MONITORING AN ISP TRANSITION

We traced a path with `tcptraceroute` from our university to an Israeli website once in two minutes from Feb. 26 to Mar. 18, 2010, which we knew contained two network switches caused by our ISP transition. We use `DiffP` to compare consecutive traceroute pairs and plot the time series of the scores in Figure 2. Both switches are detected with rather big scores, because `DiffP` finds six ASes, 15 SNs, and one IP to be different. It also confirms the first divergent IP to be in the AS which caused the change. Moreover, `DiffP` detects a load-balancer which results in constant small but non-zero scores.

We then use the `DiffP` quantification as a distance function to cluster the traceroutes with an agglomerative, complete linkage method [2]: all four clusters have distinct performance patterns. Before the first switch and after the second one (cluster 1), the path has stable RTTs with intermittent surges, whereas the RTTs in cluster 2 are notably inflated. Moreover, the second appearance of cluster 1 has shorter RTTs than the first because of the new ISP. It also has a clearer diurnal loss pattern. Cluster 3 is correlated with high packet reordering rate due to the aforementioned load-balancer.

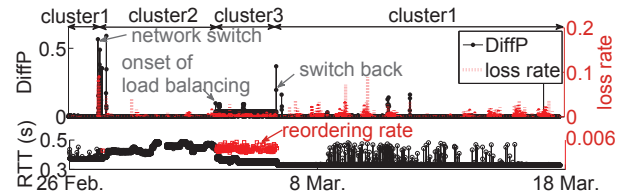


Figure 2: Correlating `DiffP` results with RTT, loss rate, and reordering rate.

5. CONCLUSIONS AND FUTURE WORK

We presented `DiffP` to avoid information entanglement in existing segregative multi-level path comparisons. We showed in an ISP transition analysis that it captures the change scope effectively, and could be used to cluster route patterns which account for the performance variations. We will use `DiffP` to troubleshoot disruptive changes due to cable failures or DoS attacks.

Acknowledgement This work is partially supported by a grant (ref. no. ITS/355/09) from the Innovation Technology Fund in Hong Kong and a grant (ref. no. H-ZL17) from the Joint Universities Computer Centre of Hong Kong.

6. REFERENCES

- [1] A. Pathak, H. Pucha, Y. Zhang, Y. Hu, and Z. Mao. A measurement study of Internet delay asymmetry. In *Proc. PAM*, 2008.
- [2] R. Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Trans. Neural Netw.*, 2005.
- [3] M. Zhang, C. Zhang, V. Pai, L. Peterson, and R. Wang. PlanetSeer: Internet path failure monitoring and characterization in wide-area services. In *Proc. USENIX OSDI*, 2004.