Using Type-of-Relationship (ToR) Graphs to select Disjoint-Paths in Overlay-Networks

Sameer Qazi and Tim Moors
School of Electrical Engineering and Telecommunications
University of New South Wales
sameerq@student.unsw.edu.au, moors@ieee.org

Abstract— **Routing policies used in the Internet can be restrictive, limiting communication between source-destination pairs to one path, when often better alternatives exist. To avoid route flapping, recovery mechanisms may be dampened, making adaptation slow. Unstructured overlays have been widely proposed to mitigate the issues of path and performance failures in the Internet by routing through an indirect-path via overlay peer(s).**

Choice of alternate-paths in overlay networks is a challenging issue. Ensuring both availability and performance guarantees on alternate paths requires aggressive monitoring of all overlay paths using active probing; this limits scalability when the number of overlay-paths becomes large. An alternate technique to select an overlay-path is to bias its selection based on physical disjointness criteria to bypass the failure on primary-path. In this paper, we show how Type-of-Relationship (ToR)-Graphs can be used to select maximally-disjoint overlay-paths.

I. INTRODUCTION

End-to-end paths in the Internet sometimes fail to deliver the Quality of Service (QoS) required by some applications. For many user-perceived performance failures/faults there exists a redundant path available which can be used to actually prevent or "mask" the fault from the end user by using quick switch-over mechanisms. One study [1] shows that for almost 80% of the paths used in the Internet there is an alternate path with lower probability of packet loss. On detection of failure on primary direct-path, the Internet switches to alternate direct-paths learnt through the Border Gateway Protocol (BGP). Despite being highly scalable, BGP only addresses reachability, without similar guarantees for QoS, and in the event of a failure uses a trial and error method to investigate each path in turn. When the direct-path between two hosts fails, overlay networks can quickly establish an indirect-path through intermediate host/s. It is found that majority of direct-path failures can be bypassed by an indirect-path through a single intermediate overlay-host (*one-hop* overlay path) [14] (Figure 1a).

Overlay-networks will only improve service if the chosen alternate-path is not affected by the failure on the primary-path. This will likely be true if the alternate-paths are disjoint from the primary. In this paper, we present an approach for selecting disjoint paths using Type-Of-Relationship (ToR) graphs; the idea is the selection of candidate indirect-paths through overlay-hosts based on maximal-disjointness from the direct-path. We validate our findings using real-world Internet-data from the Active Measurement Project (AMP) [2] to quantify the benefits of choosing paths that are disjoint in terms of Autonomous Systems (ASs) they traverse.

Section II describes ToR-graphs briefly; In Section III we present a greedy-approach for finding maximal AS-disjoint overlay-paths. In Section IV we evaluate the performance using real-world Internet-data. Section V discusses related work and Section VI summarizes the key findings of the paper.

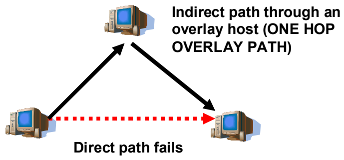

Figure 1a. Indirect-path through overlay-host when direct-path fails.

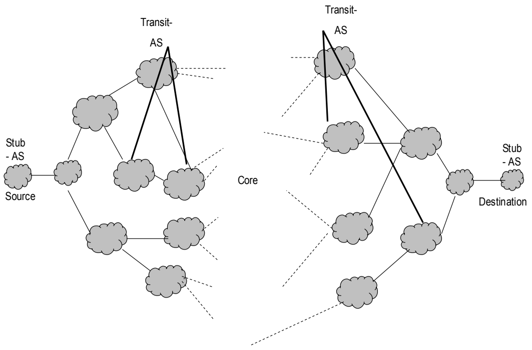

Figure 1b. Physical paths between source-destination at AS level topology.

II. ToR (TYPE-OF-RELATIONSHIP) GRAPHS

The Internet is composed of a large number of autonomous networks (ASs). Each AS is independently administered. To route a packet from one host to another it must pass via several different ASs. ASs can be characterized into two broad categories, transit ASs and stub ASs (Figure 1b). Stub-ASs are located on the edges of the Internet and typically have few connections to neighboring ASs (usually one, perhaps a few if multi-homed) whereas transit ASs usually have more connections to neighboring ASs. Each sub-network learns about global reachability to different hosts in the network by exchanging route advertisements with immediate neighbors.

Gao [3] first showed that besides the 'generic' transit-stub architecture, three dominant types of commercial-relationships occurred between ASs, namely customer-provider (*C-P*), peer-peer (*P-P*) and sibling-sibling (*S-S*). Customers depend on their respective provider networks for connectivity (the providers acting as a transit for them), usually in exchange for a fee. Peers (and siblings) are networks which are similar in scope and can exchange traffic (destined for each other's customers) between each other without a fee, for mutual benefit. *C-P*, *P-P* and *S-S* relationships are all relative in that a particular AS can have different relationships with different adjacent ASs. Gao [3] finds that the percentage of *C-P* , *P-P* and *S-S* relationships are roughly 90.5, 8 and 1.5 respectively.

Gao [3] also showed that the Internet uses "valley-free" paths between hosts which are defined by policies. The term "valley-free" refers to the hierarchy formed by customer-provider relationships. Traffic is permitted to pass up the hierarchy (from customers to their providers) at the source end of the path, but can only pass down the hierarchy in order to approach the destination; a provider cannot use one of its customers to connect to another provider, since that would form a valley. This favors the commercial-relationships between providers and customers so as to: (a) maximize the provider profit; and (b) avoid unnecessary routing loops. Figure 2 shows some examples of valid valley-free paths.

Type-of-Relationship (ToR) graphs [4,5] with vertices representing ASs show the customer/provider/sibling relationship between adjacent ASs, using directed edges for *C-P* relationships and undirected edges for *P-P* and *S-S* relationships [3]. For consistency, and without loss of generality, *P-P* and *S-S* relationships can be represented by two directed edges by introducing a virtual-provider node in between them [5]. We adopt this technique to map *P-P* and *S-S* edges in the ToR-graph (Figure 3). Note the ToR graph is a directed AS graph only depicting the relationship between ASs (*C-P*, *P-P* or *S-S*) unlike the undirected AS graphs used in [7] where an edge between two ASs depict physical path-delay of inter-AS links.

C-P, *P-P* and *S-S* relationships are never explicitly revealed because of commercial-agreements. However, by reading BGP dumps one can access AS paths (described above) which can help in inferring the type-of-relationships between adjacent AS-pairs using simple intuitive rules specified by the valley-free routing model. For example, previous works [3,4,6] identify valley-free paths as those having either (a) an uphill path, a *P-P* edge, and a downhill path in order; or (b) an uphill path and a downhill path in order (Figure 2).

Existing research finds that intuitive approaches like Earliest Divergence Rule [15] can help in finding disjoint-paths using the knowledge of AS paths between hosts (through *trace-routes*). We find that only by mapping such AS information into a ToR-graph we can use more elegant algorithms for computation of AS-disjoint paths that can give non-negligible improvement over such approaches.

III. MAXIMALLY-DISJOINT PATH COMPUTATION IN TOR-GRAPHS USING GREEDY APPROACH

A. Finding Valley-Free Edge-Disjoint Paths

To avoid a failure affecting a path, we need an alternate path which is physically-disjoint from this primary path; such paths can be either vertex-disjoint or edge-disjoint, where paths do not share edges. Our focus is on computing edge-disjoint valley-free paths in the ToR-graph, since this problem is shown to be solvable in polynomial-time while the corresponding vertex variant of the problem is NP-hard [5].

Computing all mutual edge-disjoint paths simultaneously between all possible pairs of vertices $v \in V$, in a graph $G=(V,E)$ is a NP-complete problem [8]. However, if we are only interested in computing edge-disjoint paths between *two* nodes s and t, then the problem becomes tractable [8].

To search for valley-free edge-disjoint paths in a ToR graph, Erlebach et al [5] proposed a two-layer graph (H), constructed from a ToR-graph $G=(V,E)$ and $s,t \in V$ (see Figure 3). H is a directed graph obtained by making two copies of the original graph G, called the lower and upper layer. In the upper layer all edge directions are reversed. Every node in the lower layer is connected with 'n' artificial edges to the corresponding copy of that node, denoted by v', in the upper layer. These edges are directed from v to v'. The justification of Erlebach et al's two-layer graph is as follows, and comes from the previously stated view of valid valley-free paths as being the concatenation of a set of forward edges (uphill-path) and a subsequent set of backward edges (downhill-path). A valid path $p=v_1,\ldots,v_r$ in G with $v_1=s$ and $v_r=t$ is equivalent to a path in the directed graph H in the following way. The forward part of p, i.e. all edges $(v_i,v_{i+1}) \in p$ that are directed from v_i to v_{i+1}, is routed in the

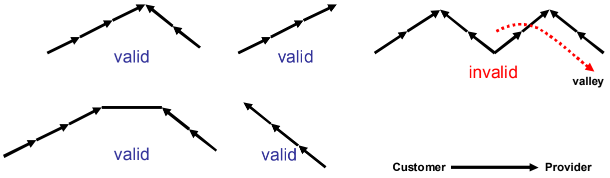

Figure 2. Example of valid and invalid valley-free paths in ToR-graphs

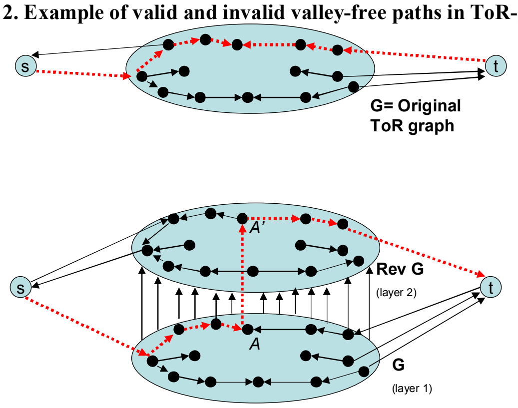

Figure 3. Example of valid valley-free path in the original ToR-graph (G) (top) and the relaxation using the 2 layer model (bottom).

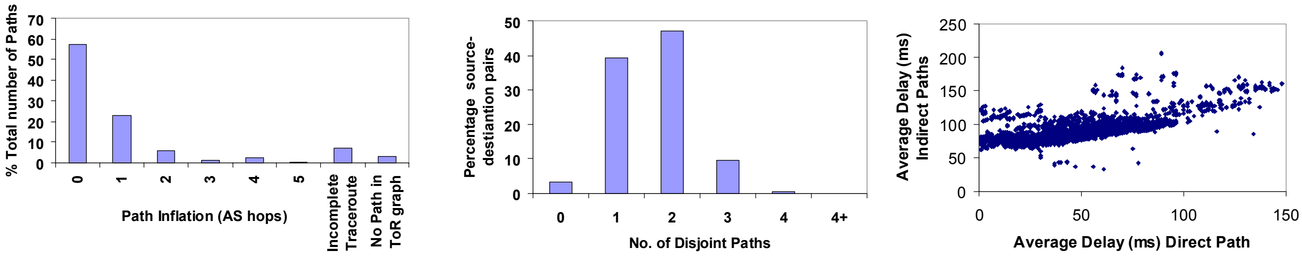

Figure 4(a). (left)Inflation on paths between AMP hosts (AS-hops). Figure 4(b). Number of disjoint paths for source-destination pairs using ToR-graph. Figure 4(c). Relationship of average delay on direct and indirect-paths (AMP Dataset 30 Jun 06)

lower layer. Then there is a possible switch to the upper layer (there can be at most one such switch, enforced by directed artificial links between G and its reverse). The backward part of *p* is routed in the upper layer (see Figure 3). The *n* parallel artificial edges of type (*v*,*v'*) going from each node of the lower layer to its corresponding copy in the upper layer have been added to *H* so as to ensure that an arbitrary number of paths arising from edge-disjoint paths in *G* can switch from the lower layer to the upper layer. Erlebach et al [5] show that the two-layer model yields an optimal solution to finding the maximum number of valley-free edge-disjoint paths.

B. *Finding Maximally-Disjoint Paths*

To compute such maximally-disjoint paths using the ToR-graph, we use a greedy-approach. We briefly formalize our technique for searching edge-disjoint valley-free paths between source-destination hosts. Given a directed ToR-graph $G=(V,E)$ (where $v \in V$, $e \in E$) and two distinguished nodes s (the source) and t (the destination); the search-algorithm starts out with an empty solution set S and in each subsequent iterations, the shortest-path is found between s and t. Once a path p_x is found, it is added to S and the edges used in the current path are deleted and the process is repeated on the remaining graph until no further s-t path can be found. The path found in the last iteration is taken as the candidate path which is maximally disjoint from the primary (direct) route between hosts.

Since, Internet selects shortest paths between hosts, by eliminating shortest paths first, the path found in the last iteration is most likely to be maximally disjoint from the primary path and identifies ASs not used on the direct path. Computing the shortest path in the AS graph to approximate the shortest Internet path is challenging problem as argued by [7]. This is due to the facts that sometimes Internet does not select shortest paths due to BGP policies and that there may be more than one shortest-path with same number of AS hops. However, these issues can be resolved as suggested in [7] by using additional criteria such as making use of the fact that AS-paths are transitive and symmetric. Since, in this paper the ToR graph is constructed using only AS-paths between overlay end-hosts instead of reading BGP dumps, the ToR-graph is sparse and hence the number of paths between any hosts is not large. Also, note the aim of the greedy-approach is **not** to predict the shortest-path between hosts likely used by Internet but on the contrary to only identify the ASs on the most-disjoint valley-free path.

IV. PERFORMANCE EVALUATION

A. *Methodology and data sets used to construct ToR-graph*

We use path and delay measurements collected between 62 Active Measurement Project (AMP) [2] hosts from North and South America during two 24 hr periods on June 30 and August 31, 2006. These datasets provide about one round-trip time (RTT) delay measurement for each pair of nodes per minute, and IP-trace-route information obtained around once every ten minutes. While the aim of an overlay network may only be to optimize the one-way delay, which may differ for different directions due to asymmetric Internet paths, two-way delay-measurements, such as RTTs, have been shown [10] to be strongly correlated (with a correlation-coefficient of 0.87) to one-way delays, and so form a reasonable basis for inferring one-way delays.

To construct ToR-graph, we first identify all ASs used by paths between all possible AMP hosts (3782=62*61). We identified a total of 4400 unique IP-addresses from the IP trace-route information. To map these IP addresses to AS numbers we use the IP-to-ASN Whois Service from Cymru [11], which provides mappings for user-specified dates using GNU netcat utility[12]. Using the results from this service we identified a total of 275 unique ASNs. To find the relationships between these ASs (*C-P*, *P-P*, or *S-S*) for constructing ToR-graph; we use the AS-relationships data from CAIDA [13].

B. *Physical path characteristics inferred from ToR-graph*

Since we aim to use a heuristical approach for finding maximally-disjoint overlay paths, we first look at AMP data to evaluate the effectiveness of our proposed techniques. To see the impact of routing-polices on path-inflation; we compute the shortest-paths between AMP hosts in the ToR-graph and compare it with the actual number of AS hops on the direct-path using the trace-route information from the dataset. We find that while the majority (57%, see Figure 4a) of paths between AMP hosts were shortest-possible AS paths, 23% of paths were inflated by at least one AS hop and around 10% by two AS hops or more.

We also measure the total number of edge-disjoint paths found per source-destination pair (Figure-4b). Around 47% of AMP host-pairs have two and 10% AMP host-pairs have three edge-disjoint paths.

C. Performance-Evaluation of Greedy-Approach

1) Selection of Alternate Paths

The greedy-approach selects alternate-paths between source-destination pairs by ranking them on the basis of their degree-of-disjointness from direct-paths. For this, we first use the traceroute information on all possible one-hop indirect paths and compare the number of ASs which are common between the indirect path and the candidate-path selected by our algorithm.

We define degree of disjointness (σ_n) of n^{th} overlay-path as being the ratio of the number of ASs that are common in the candidate valley-free disjoint-path computed by the greedy-approach (*cdp*) and the n^{th} overlay path. We use this degree of disjointness to rank overlay paths. Thus, given the candidate-disjoint-path (*cdp*) between two AMP-hosts (*s* and *d*) by greedy-approach, using the ToR-graph as AS_{cdp}=[AS_s AS_w AS_x AS_y...AS_d] and the corresponding one-hop indirect-path between the same host-pair as $AS^n_{1\text{-}hop}$ (for the n^{th} indirect-path)= [AS_s AS_p AS_q AS_r ...AS_d], the degree-of-disjointness coefficient (σ_n) is given by (1):

$$\sigma_n = |AS^n_{1\text{-}hop} \cap AS_{cdp}| \,/\, |AS^n_{1\text{-}hop}| \qquad (1)$$

(where | X | denotes the number of elements in X.)

An alternate path *n* is selected by the greedy-approach if the partial disjointness is greater or equal to some threshold value σ, i.e. n^{th} alternate-path is selected if $\sigma_n \geq \sigma$. We report our results as we varied the value of σ from 0.2 to 0.7 (i.e. least to most disjoint paths) as most σ_n values were in this range.

Note, if the number of edge-disjoint paths in the ToR-graph between a given source-destination pair is only one (Figure 4b), the greedy-approach may actually choose the *shortest* path as opposed to more circuitous disjoint-path. Note that this **does not** invalidate the effectiveness of the greedy-approach, since even selecting a shorter-path between AMP hosts can still yield a path that is disjoint from the primary-route if the direct-path is inflated (Fig 4a). If the direct path is not inflated then greedy-approach is unable to filter one-hop indirect-paths on the basis of their disjointness from direct-path. Interestingly, we found out that for such source-destination pairs showing little or no path diversity, even the most intuitive strategy like the Earliest Divergence Rule [15] was unable to select a small number of candidate alternate-paths because a large number of alternate-paths diverged at the same AS hop. In such situations, [15] proposes the use of selecting paths based on additional path-performance criteria such as delay constraints; the focus of this paper is not to investigate such criteria; the performance is evaluated strictly under the disjointness criteria mentioned previously.

2) Predictability in Selecting Good Disjoint Paths

We measure predictability of greedy-approach as being its ability to select *good* paths (defined below) when the direct path has *bad* path metrics. We use delay-metric to rate path-quality as delay and loss-rates on paths are highly correlated [15]. We thus formally define the predictability of the greedy-approach as its ability to select good paths (as explained earlier) and defined in 2:

Predictability=Prob[*good-selected*/*total selected(good+bad)*] (2)

From the data-sets we found that if the direct-path *i* had delay D_i then the delay of majority of alternate paths ranged between D_i and $2\ D_i$ (Figure 4c), hence selecting an alternate path does not make sense unless the delay on the direct path exceeds a factor of two already. We designate a direct-path as degraded if the path-delay exceeds its average delay by a factor of two and an indirect-path as being *good* if its delay is still within a factor of two of this value during the time the direct-path stays degraded (otherwise it is considered *bad*). [15] finds that, "…which paths are good alternates to avoid delay degradations is relatively insensitive to the exact definition of delay degradation".

We next carried out simulations to analyze the fault-tolerance properties of maximally-disjoint paths when the direct-path undergoes an outage. For all AMP hosts, we observed intervals when the path between them suffered from outage or path degradation as defined earlier. We investigate which indirect-paths offer better performance during the entire period when the direct path is degraded by using the time-stamps in the RTT trace files in the AMP dataset [2].

The greedy-approach reduces the number of candidate selected paths as the disjointness threshold σ increases from 0.2 to 0.7. However, as we increase σ to 0.7, the number of candidate choices also reduces to zero for approximately 50% of samples (Figure 5). This is because the candidate-disjoint path (*cdp*) selected by the greedy-approach only potentially identifies ASs on most-disjoint overlay-paths. It is possible that many of the ASs identified are not present on an overlay path if we set σ very high.

We found that the predictability of the greedy-approach increases as we increase σ from 0.2 to 0.5; as σ is increased further the predictability of the algorithm decreases as supported by Figure 5. We plot the predictability-metric for the greedy-approach for both June and August datasets for σ=0.5 (which gave us the best results for predictability) (Figure 6). The greedy-approach gives high predictability for both datasets; having probability of success of 0.7 or higher for 80% and 60% samples for June and August datasets.

For performance comparison, we also show the results of selecting paths using the Earliest-Divergence criteria [15] where those alternate paths are considered whose AS paths separate from direct path nearest to the source. Overall we observed that selecting alternate indirect-paths on the basis of greedy-approach, not only selected fewer disjoint paths than Earliest-Divergence [15] criteria, from 60 to about 5 (σ=0.5) in 50% of the samples; but it also improved the predictability by at least 10-20% over Earliest-Divergence [15] selection.

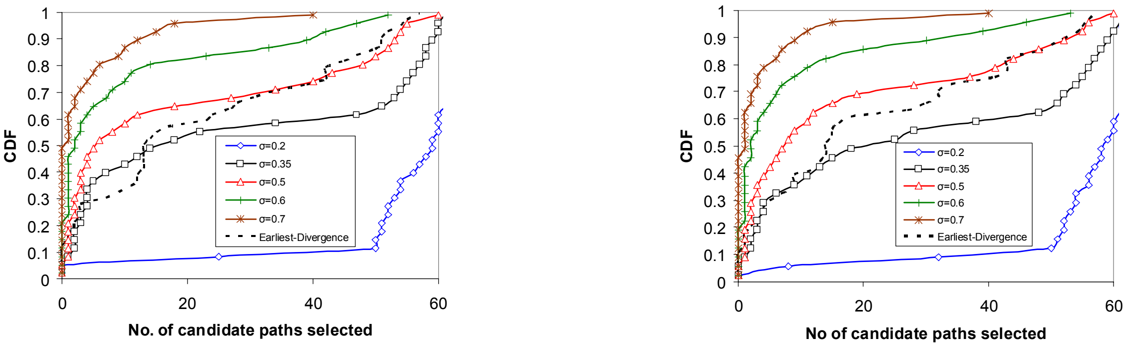

Figure 5. Number of candidate paths selected by greedy-approach for the AMP-datasets: (a) 30 Jun 06 (left) and (b) 31 Aug 06.

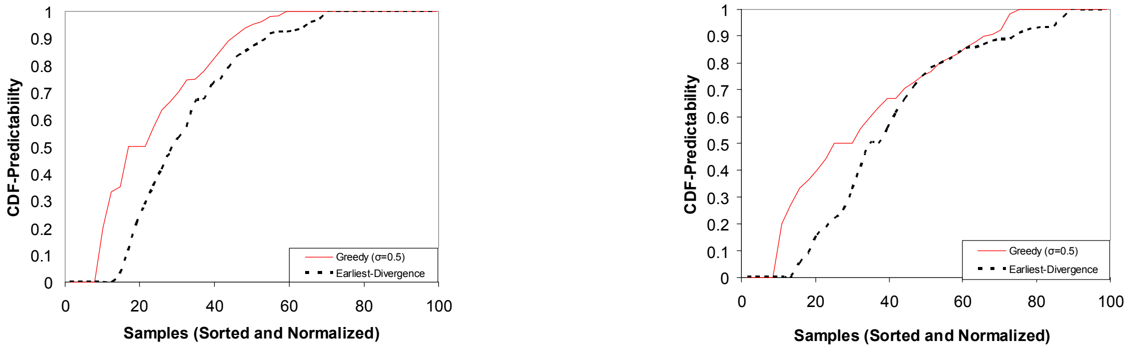

Figure 6. Predictability of greedy-approach for the AMP-datasets: (a) 30 Jun 06 (left) and (b) 31 Aug 06.

V. RELATED WORK

Finding alternate-paths to act as a backup when primary routes fail in overlay networks is previously explored in [14]. Anderson et al, designed RON to be a resilient routing tool for the Internet by implementing a small link state overlay (50 nodes). The overlay tries to find the best alternate path to the destination. The best path may be the default Internet path or an alternate overlay path. The design posed scalability problems due to extreme bandwidth requirements for active probing of all virtual overlay links and subsequent dissemination of this information using a link-state protocol. [17] shows that in most cases alternate paths can be found using at most one overlay hop but does not address the path selection problem. Topology aware approaches have been extensively studied to counter the scalability issue. [7] discusses a proposal for 'pruning' the overlay topology through removal of redundant physical links which are not likely to be selected by the overlay routing algorithm. Disjoint-Path selection in the Internet has been recently investigated in [15], which proposes selecting an indirect-path through an overlay node whose AS-level path digresses most-quickly from the direct-path.

VI. CONCLUSION

This paper presents the first analysis of computing maximally-disjoint paths in overlay networks using ToR graphs. Disjoint path computation can be used as an offline-heuristic to supplement measurement-based approaches [14] which are not scalable, or for alternate indirect-path computation when the direct path between two hosts is affected by a performance failure or an outage. We proposed and analyzed the performance of an intuitive greedy approach for computation of such disjoint-paths using real world Internet datasets. Our results show that such heuristics can be used to select alternate paths to bypass path outages or degradations.

REFERENCES

[1] S. Savage et al., "The End-To-End Effects Of Internet Path Selection," SIGCOMM '99

[2] Active Measurement Project (Amp). See http://watt.nlanr.net/

[3] L. Gao, "On Inferring Autonomous System Relationships In The Internet," IEEE/ACM Trans. Netw., Vol. 9(6):733-45, 2001

[4]G. Di Battista et al., "Computing The Types Of The Relationships Between Autonomous Systems", IEEE INFOCOM 2003

[5] T. Erlebach et al., "Cuts And Disjoint Paths In The Valley-Free Path Model", Proc. Workshop On Combinatorial And Algorithmic Aspects Of Networking (CAAN 04)

[6] J. Xia And L. Gao, "On The Evaluation Of AS Relationship Inferences", Proc. IEEE Global Telecommunications Conference, (GLOBECOM), 2004

[7] A. Nakao et al., "Scalable routing overlay networks," SIGOPS Oper. Syst. Rev., 40(1):49-61, 2006

[8] J. Kleinberg, "Approximation Algorithms For Disjoint Paths Problems, Phd Thesis," In Dept. Of EECS MIT 1996

[9] Y. Zhang And N. Duffield, "On The Constancy Of Internet Path Properties," Proc. ACM SIGCOMM Workshop On Internet Measurement, (IMW)2001.

[10]T. Rakotoarivelo et al., "Enhancing QoS Through Alternate Path: An End-To-End Framework ", Proc. Intl. Conf. On Networking (ICN), 2005

[11]Cymru IP-To-ASN Whois Service. http://www.cymru.com/

[12]GNU Netcat. See http://netcat.sourceforge.net

[13]CAIDA AS-Relationships-Dataset, http://www.caida.Org/Data/Active/AS-Relationships/

[14]D. Andersen et al.,"Resilient Overlay Networks", Proc. ACM Symp On Operating Systems Principles (SOSP) 2001

[15]T. Fei et al., "How To Select A Good Alternate Path In Large Peer-To-Peer Systems?", IEEE INFOCOM 2006

[16] Jon Kleinberg and Eva Tardos, "Algorithm Design", Addison Wesley, ISBN 0-321-29535-8

[17] K. Gummadi et al., "Improving the Reliability of Internet Paths with One-hop Source Routing", USENIX Symp. Operating System Design and Implementation (OSDI), pp 183-98, 2004