Topology Modeling via Cluster Graphs

Balachander Krishnamurthy and Jia Wang

Abstract— **Several recent studies have focused on generating Internet topology graphs. Topology graphs have been used to predict growth patterns of prefixes and traffic flow as well as for designing better protocols. Internet topology graphs can be studied at either inter-domain level or router level. For some applications, inter-domain level topology graph is too coarse, while router level topology graph may be too fine-grained. We introduce *cluster graphs* as a way of modeling Internet topology at an intermediate level of granularity and compare it against inter-domain and router graphs.**

I. Introduction

The Internet can be decomposed into connected *autonomous systems*[1] (ASes) that are under separate administrative control. Rapid growth of the Internet has necessitated the understanding of network topology. Recent studies have attempted to create Internet topology graphs at both AS-level and router-level. Researchers have been using topology graphs to gain better understanding of how the Internet behaves under various traffic patterns and to design protocols that take advantage of the underlying network topology. Internet topology graphs are also used in evaluating the performance of applications such as Web caching, resource replication, and content distribution networks (CDNs). Although topology graphs are used heavily in Internet research, neither kind of the topology graphs are suitable for most applications. The AS graph is too simple and may not reflect the actual properties of the underlying networks it represents. The router graph is expensive to generate, too fine-grained, and not feasible for most applications. We introduce a new way to model the Internet topology using a *cluster graph*. A cluster graph models topology at an intermediate level of granularity, i.e., it is more fine-grained than an AS graph and more coarse-grained than a router graph. We use the network-aware clustering technique [9] to group routers and hosts into clusters based on IP addresses. In a cluster graph, a node represents a cluster of routers and hosts, and an edge represents an inter-cluster connection. We first show that ASes are too coarse-grained to model the network proximity of IP addresses. Then, we show that cluster graphs can be constructed efficiently and provide a better model of the Internet topology.

The authors are with AT&T Labs–Research, Florham Park, NJ, USA. email:{bala, jiawang}@research.att.com Contact author: Jia Wang, tel: 973-360-8324, 180 Park Avenue, Florham Park, NJ 07932

[1]The term of *autonomous system* (AS) and *domain* are used interchangeably in this paper.

II. Internet topology graphs

Recent studies attempt to model the Internet topology at two levels of granularities. At the *inter-domain* level, an AS graph is generated to model the Internet topology [15], [3], [11], [10], [8]. In an AS graph, a node represents a single domain (AS) and an edge represents an inter-domain connection. At the *router* level, a node in the router graph represents a router [13], [1], [2], [5], [15] and an edge represents the adjacency of two routers. Approaches to generating AS graphs can be grouped into three categories:

1. *AS Path-based AS graph.* The AS graph[2] is derived from the *AS Path* information in BGP *Update* messages [6] or in BGP tables [14]. Each element of an *AS Path* defines a node and each successive pair of domains in the *AS Path* represents the endpoints of an edge. Although simple, this approach may not yield a complete AS graph due to the unavailability of the complete BGP routing information. Also, the AS graph derived from BGP *AS Path* information does not necessarily reflect the actual policy routing paths of data packets. Because inter-domain routing is controlled by BGP, connectivity does not imply reachability [4].

2. *Traceroute-based AS graph.* Here, the AS graph is derived [15] through extensive *traceroute* probings on IP addresses and lookups in BGP routing tables. It first uses *traceroute*-like probing technique to explore the IP address space for addressable routers and hosts and infers a router graph. Then, an AS graph is computed by mapping each IP address to the AS responsible for routing it, i.e., the origin (end-of-path) AS for the best match IP prefix of this address in BGP routing tables. The main advantages of this approach is that it reflects the actual policy routing paths. However, this approach relies on extensive network probing to yield a reasonably complete graph and may not be feasible for most applications.

[2]Note that the AS graph derived from *AS Path* information extracted from BGP (Border Gateway Protocol [7]) *Update* messages may be different from the one derived from the *AS Path* information extracted from the BGP tables. The AS graph derived from BGP *Update* messages usually has more edges. The advantage of using BGP routing tables is that they are much easier to collect than *Update* messages

3. *Synthetic AS graph.* A synthetic AS graph is generated based on observed characteristics such as power law properties [3], [11], [10], [8], [12]. The advantages of such topology generators is that they can generate AS graphs that obeys some desired characteristics. It is useful for protocol design and evaluation. However, the problem of generating realistic Internet topology is yet to be solved. The synthetic AS graphs that obey power laws are different from the real inter-domain topology, for example.

Although AS graphs have been used in many applications as a model of the Internet topology, they are too simple to capture the real network topology and characteristics. ASes are too coarse-grained to model the network proximity of IP addresses. For example, the correlation between the AS hop count and the corresponding end-to-end latency is weak. If a private AS number is used, a single node in the AS graph may correspond to multiple domains that could be far away from each other.

III. Cluster graph

In prior work [9], we introduced network-aware clustering as a way of identifying a set of IP addresses that are with high probability under common administrative control and topologically close together. We use the term *cluster* to denote such a group of IP addresses. We group routers and hosts into clusters and propose a new model of the Internet topology—*cluster graph.* A cluster graph is an undirected graph with a node representing a unique cluster of routers and hosts. The edge connecting two nodes represents inter-cluster connection. We construct cluster graphs either based on real data, such as BGP tables and *traceroute* results, or based on some observed properties of clusters, such as power laws. We propose three techniques for generating cluster graphs.

1. *Hierarchical cluster graph.* The cluster graph has two levels. The higher level of the cluster graph is the same as the AS graph derived from the *AS Path* information in BGP tables or *Update* messages. For each node (i.e., AS) in the top level graph, the lower level of the cluster graph is a mesh of clusters belonging to the AS. The mapping between clusters and ASes can be computed using information in the BGP tables.

This simple method can be viewed as an improvement to the AS graph by modeling the size of ASes. Both the number of clusters belonging to an AS and the size of the clusters are metrics to measure the size of ASes. The size of a cluster (AS) is defined as the total number of potentially usable IP addresses belonging to the cluster (AS). The cluster graphs obtained using this method are less realistic than the real Internet topology because it does not model the connectivity among clusters.

2. *Traceroute-based cluster graph.* We first choose a few IP addresses at random from each cluster and run *traceroute* to them from various places (e.g., *traceroute* gateways). Then, we group all the routers and end hosts extracted from the *traceroute* results into clusters and derive a *cluster path* between each pair of the end hosts. Based on the cluster paths, we generate a cluster graph whose nodes are the union of the sets of nodes on all the cluster paths. For each adjacent pair of nodes on each cluster path, we add an edge in the cluster graph connecting the corresponding pair of nodes before removing all redundant edges.

We further derive corresponding AS paths based on the mapping between clusters and ASes. An AS graph can be derived based on AS paths using a technique similar to constructing a cluster graph. In addition, based on the mapping between clusters and ASes, each node in the AS graph can be expanded as a group of connected clusters belonging to the same AS to form a cluster graph, where the connectivity between each pair of clusters within an AS is determined by the corresponding cluster paths.

One may derive a router graph based on the *traceroute* paths. Unlike constructing cluster graphs, constructing a useful router graph requires extensive *traceroute*-like probing on the Internet IP address space and techniques to resolving interface aliases (interfaces belonging to the same router). We only need to probe very few IP addresses in each cluster to construct a cluster graph. This is more useful when partial topology graph is needed in some applications such as peer-to-peer applications. We can probe a few sample IP addresses from the interested clusters to construct a cluster graph, whereas extensive probing is needed to construct a router graph for the IP addresses of interest. If the router graph is already available the corresponding cluster graph can be easily constructed. We collapse a group of router nodes belonging to the same cluster into a cluster node and remove all the edges that connect to two router nodes belonging to the same cluster as well as the redundant edges between the cluster nodes.

3. *Synthetic cluster graph.* Similar to existing AS graph generators, we generate cluster graphs based on some observed characteristics of cluster topology, such as a power law. Useful metrics include node degree distribution, rate of spreading and eigenvalues of the graph, and weights of the node (e.g., diameter of clusters and size of clusters).

IV. Experiments

We now present early results from ongoing experiments that examines the feasibility of cluster graphs. We first examine the relationship between clusters and ASes and show that the AS-level super-clustering is too coarse-grained. Next, we compare cluster graphs with corresponding AS graphs and router graphs.

A. Super-clustering

We group clusters that belong to the same AS into a *super-cluster* by identifying the ASes originating the prefixes of the clusters. Clusters whose prefixes are originated by the same AS are grouped together into a super-cluster and identified by its AS number. We used BGP routing tables collected in May 2001 at the same set of locations as in [9], and extracted 114,620 unique IP address prefixes originating from 10,834 unique ASes. The number of prefixes originated by an AS ranges from 1 to 5,623 with an average of 11. More than half of the ASes originate more than 1 prefix.

We use a large portal Web site's server log obtained in March 2001 with 104,018,140 requests issued by 7,652,670 unique clients. All the client IP addresses are grouped into 15,789 clusters. The number of requests issued from a cluster varies from 1 to 1,432,645, the number of clients in a cluster varies from 1 to 57,478. We sort the clusters in the reverse order of the number of requests issued from within a cluster and retain the top 3,000 busy clusters issueing a total of 73,068,844 requests (70% of total requests). We group these busy clusters into 1,250 super-clusters. There are 436 ($\sim 35\%$ of the total) super-clusters that contain multiple busy clusters. The average number of busy clusters in a super-cluster is 2.4.

The AS-level super-clusters are too coarse-grained to model the network proximity of IP addresses. An AS can be very large and may contain several smaller entities which are separately administered, that is, an AS may contain multiple clusters. For example, the super-cluster AS 1221 contains 8 busy clusters (Table I). They are unlikely to be under the same administration and should be treated as separate entities. In addition, it is not a one-to-one mapping between busy clusters and ASes. An AS may contain multiple clusters. Super-clustering builds accurate many-to-one mapping between clusters and ASes.

Cluster prefix/netmask	Common name suffix
139.130.0.0/16	wnise.com
139.134.0.0/16	tmns.net.au
192.148.160.0/24	telstra.com.au
203.32.0.0/14	ocs.com.au
203.36.0.0/16	trickysoft.com.au
203.38.0.0/16	panorama.net.au
203.0.0.0/10	geelong.netlink.com.au
203.0.0.0/12	iaccess.com.au

TABLE I

THE BUSY CLUSTERS IN SUPER-CLUSTER AS 1221.

B. Cluster graphs

To evaluate how well the cluster graph models the Internet topology, we compare the *traceroute*-based approaches to constructing router graph, cluster graph, and AS graph. We use the same server log to illustrate the experimental results. We sample 99 client IP addresses randomly choosing one from each of the top 99 busy clusters. We run *traceroute* from a single source to each of these 99 IP addresses and resolve the paths. The average length of paths (in hop counts) towards the sampled IP addresses is 16. A set of 748 unique IP addresses are obtained from the *traceroute* source, the 99 sampled IP addresses, and all the intermediate hops of the 99 *traceroute* paths. We also obtain 850 unique edges connecting a pair of adjacent routers from the 99 *traceroute* paths.

We group all of the 748 IP addresses into 241 distinct clusters. For the path between the *traceroute* source and each sampled IP address, we collapse the nodes/hops belong to the same cluster into a cluster-node. We remove edges connecting nodes belonging to the same cluster to derive a *cluster path* for the corresponding sampled IP address. The average length of the cluster paths is 8$\sim$9. For example, running *traceroute 130.75.2.24* yields router path[3] shown in Figure 1. We group *traceroute* source and IP addresses of the 15 hops along the path into 7 clusters. Thus, we derive a cluster path from *traceroute* source to 130.75.2.24 (Figure 1). We also observe that there is a loop on the cluster path. It is easy to detect such loop in a cluster path and infer the actual routing path of data packets. However, it is difficult to identify such a "detour"[4] based on router interface IP addresses and infer the actual routing path of data packets. These 99 cluster paths from a single *traceroute* source towards the sampled 99 IP addresses form a cluster graph that consists of 241 nodes and 232 links. The average node degree is 1.9. Compared to the *traceroute* paths, the cluster graph has a lot fewer nodes and edges.

We construct AS graph based on the same data obtained from *traceroute* using the technique presented in [15]. The top 99 busy clusters consists of 1,191,084 unique IP addresses. We ran extensive *traceroute*s to these IP addresses and computed a router graph and then construct an AS graph based on the router graph. The resulting AS graph has 180 nodes and 202 links. The average AS path length is 7 and the average node degree is 2.25. We observe that

[3] The first few hops inside local domain are not shown. The destination 130.75.2.24 is the last hop (the 15th hop) on the path.

[4] We use the term *detour* instead of *loop* because it may not necessarily be a loop in the router lever path.

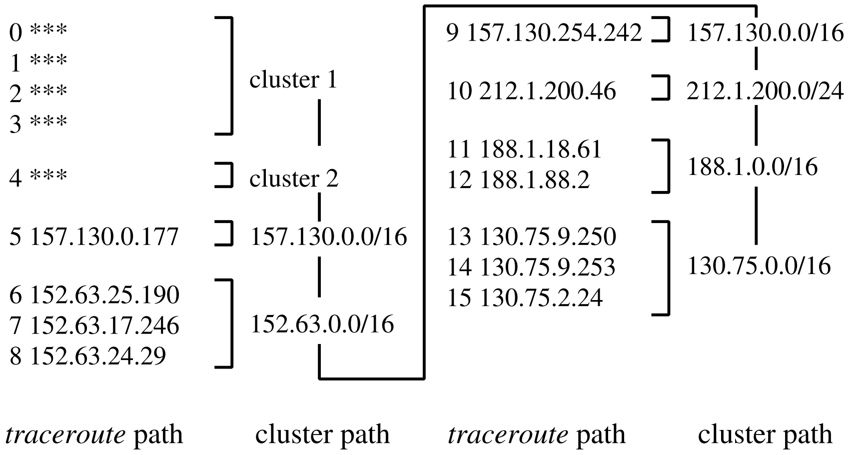

Fig. 1. The cluster path towards 130.75.2.24.

the cluster graph has 34% more nodes and 15% more edges than the AS graph. The average degree of the nodes in the cluster graph is 15% less than that in the AS graph. We found that the variance of the cluster diameters is smaller than that of the ASes. We observe that the correlation between cluster hop counts and end-to-end router hop counts is stronger than that of AS hop counts. This observation also holds for end-to-end latencies.

The cluster graph can be constructed efficiently. The *traceroute*-based approach to constructing router graph and AS graph is too expensive. Compared to the 1,191,084 *traceroute*s required by the current approach, our approach to constructing cluster graph requires only 99 *traceroute*s. The cluster graph is also more stable. We repeated our experiments by varying (i) the *traceroute* source from 6 geographically distributed *traceroute* gateways [16]; (ii) the number of sampled clusters from 100 to 10,000; (iii) the number of *traceroute*d IP addresses in each sampled clusters from 1 to 10% of the number of IP addresses in a cluster. We observed that the percentage of IP addresses that remain in the same cluster over a period of time is quite high. The cluster paths and graphs are stable.

V. Conclusion

We examined different Internet topology models and introduced a new model based on cluster graphs. We compared the Internet topology graphs at the three level of granularities: inter-domain, cluster, and router level. We show that, while AS graph is simple and easy to obtain, it is too coarse-grained to model the network proximity of IP addresses. The cluster graph is less complicated and more stable than the router level topology, can be obtained as easily as an AS graph while providing more fine-grained information than an AS graph, yet capturing the Internet topology.

References

[1] H. Burch and B. Cheswick. Mapping the Internet. In *IEEE Computer*, April 1999.
[2] K. C. Claffy and D. McRobb. Measurement and visualizatiion of Internet connectivity and peformance, http://www.caida.org/tools/skitter/.
[3] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the Internet topology. In *Proceedings of ACM SIGCOMM'1999*, pages 251–262, August/September 1999.
[4] L. Gao. On inferring autonomous system relationships in the Internet. In *Proceedings of Global Internet 2000*, November 2000.
[5] R. Govinda and H. Tangmunarunkit. Heuristics for Internet map discovery. In *Proceedings of IEEE Infocom*, April 2000.
[6] Ramesh Govindan and Anoop Reddy. An analysis of Internet inter-domain topology and route stability. In *Proc. 16th IEEE INFOCOM*, April 1997.
[7] B. Halabi. *Internet Routing Architectures*. Cisco Press, 1997.
[8] C. Jin, Q. Chen, and S. Jamin. Inet: Internet topology generator. Technical Report CSE-TR-433-00, University of Michigan, September 2000.
[9] B. Krishnamurthy and J. Wang. On network-aware clustering of web clients. In *Proceedings of ACM SIGCOMM*, August 2000.
[10] A. Medina and I. Matta. BRITE: A flexible generator of Internet topologies. Technical Report BU-CS-TR-2000-005, Boston University, January 2000.
[11] A. Medina, I. Matta, and J. Byers. On the origin of power laws in Internet topologies. *ACM Computer Communication Reciew*, 30(2), April 2000.
[12] Christopher R. Palmer and J. Gregory Steffan. Generating network topologies that obey power laws. In *Proceedings of GLOBECOM'2000*, November 2000.
[13] J. J. Pansiot and D. Grad. On routes and multicast trees in the internet. In *ACM Computer Communication Review*, January 1998.
[14] L. Qiu, V. N. Padmanabham, and G. M. Voelker. On the placement of web server replicas. In *Proc. 20th IEEE INFOCOM*, 2001.
[15] H. Tangmunarunkit, R. Govindan, D. Estrin, and S. Shenker. The impact of routing policy on Internet paths. In *Proc. 20th IEEE INFOCOM*, April 2001.
[16] Traceroute.org. `http://www.traceroute.org`.