# Cheleby: A Subnet-level Internet Topology Mapping System

Hakan Kardes and Mehmet Gunes
Department of Computer Science and Engineering
University of Nevada, Reno
hkardes,mgunes@cse.unr.edu

Talha Oz
Department of Computer Science
George Mason University
toz@cse.unr.edu

***Abstract*—Understanding the topological characteristics of the Internet is an important research issue as the Internet grows with no central authority. Internet Topology mapping studies help better understand the dynamics of the Internet backbone network. Knowing underlying topology, researchers can develop new protocols and services or fine-tune existing ones. In this paper, we first discuss issues in subnet-level Internet topology mapping and present approaches to handle them. Then, we introduce *Cheleby*, an integrated Internet topology mapping system. Cheleby, first, dynamically probes every observed subnetwork in the Internet using a team of PlanetLab nodes around the world. Then, it utilizes efficient algorithms for resolving subnets, IP aliases, and unresponsive routers in collected data to provide subnet-level topologies. Different from current topology mapping systems, Cheleby not only samples the Internet topology but also processes the collected data to build more complete maps. Sample topologies are provided at `http://cheleby.cse.unr.edu`.**

***Index Terms*—Internet Measurement, Router-level, Topology Sampling.**

## I. Introduction

Internet, the largest man made complex network, is a web of interconnected backbone networks over which thousands of small and medium size Autonomous Systems (ASes) connect individuals, businesses, universities, and agencies. Internet is a spontaneously growing complex system whose large-scale structure is affected by many interacting units aimed at optimizing local communication efficiency without a central authority. While the building blocks of the Internet, its protocols and individual components, have been subject to intensive studies, the immense global entity has not been precisely characterized.

The Internet's global properties can not be inferred from the local ones as it is composed of networks engineered with large technical diversity and range from small local campuses to large transcontinental backbone providers [1]. Additionally, the Internet evolves with the interplay between cooperation, so that the network works efficiently, and competition, so that providers earn money. Routers and links are added by competing entities according to local economic and technical constraints where topology information is kept confidential due to various privacy and security concerns [2]. Combination of all of these factors results in a general lack of understanding about the topological characteristics of the Internet. The confidentiality of network topology introduces challenges for the research community and requires them to infer the topology by using measurement probes.

Need for Internet measurements arises due to commercial, social, and technical issues and provide insight into network topology, routing, protocols, and applications. Topological analysis of the Internet is needed to develop network planning, optimal routing algorithms, and failure detection measures [3]. Researchers test new protocols and systems using simulations or emulations, but more realistic results can be obtained when real topologies are fed to the models [4], [5]. Additionally, network anomalies can be identified using topology measurements [6]–[8]. Analyzing Internet topology also provides insight into current trends. For instance, Gill et al. pointed out that content providers are deploying their own networks which has a flattening effect on the hierarchical AS model [9]. Similarly, evolution of the Internet topology can be analyzed to predict future growth [10].

The research community has been conducting numerous Internet measurement studies to answer various questions on the functional and topological characteristics of the Internet. Internet measurement studies require availability of representative topology maps. Depending on the nature of measurement study, researchers may use different types of topology maps including AS level [11], [12], point-of-presence (POP) level [13], [14], router level [15], link level [16] or IP address level maps [17]. A POP level topology map is often the most detailed information that ASes make publicly available, if at all, about their network [2].

In general, Internet topology measurement studies consist of three phases: (1) topology sampling, (2) topology construction, and (3) topology analysis. Inaccuracies in the first two processes may significantly affect the accuracy of the observations or results obtained in the measurement study [18]–[21]. In this paper, we first briefly define the issues in the Internet topology mapping and proposed approaches in earlier studies. We especially focus on the topology sampling and construction processes that can significantly affect observations about the underlying network [18]–[22]. Then, we present Cheleby, an Internet topology mapping system that provides insight into the Internet topology by taking continuous snapshots of the underlying networks. The system utilizes efficient algorithms to process large scale data-sets collected from distributed vantage points and provides accurate topology graphs at subnet level.

Cheleby topology mapping system, shown in Figure 1, runs on a server which actively manages PlanetLab nodes as its

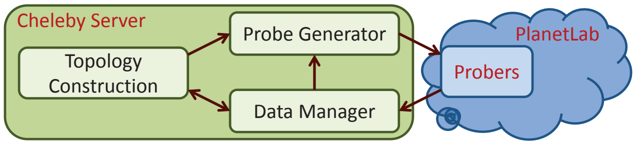



Fig. 1: Cheleby System Overview

monitors to collect topology information from geographically diverse vantage points. The server instructs monitors to collect partial path traces and perform other probing activities. Cheleby then resolves subnets, IP aliases and unresponsive routers within the collected raw data to construct the network graph corresponding to the sampled network. Incorporating enhanced resolution algorithms, Cheleby provides comprehensive topology maps at the subnet-level.

In Section II, we provide brief definitions of some issues in Internet topology mapping studies. In Section III, we briefly present major Internet topology mapping systems. Then, in Section IV, we present overview of the Cheleby system and experimental results with various system parameters. In Section V, we discuss topology construction steps in Cheleby and present some experimental results. Finally, in Section VI, we conclude the paper and provide a brief overview of future work to enhance the Cheleby Internet topology mapping system.

## II. Background on Topology Mapping

In this section, we briefly clarify some issues in Internet topology mapping studies.

**Sampling Bias:** An important issue in topology collection is to eliminate sampling bias [20], [23]. Since there are limited number of vantage points and a large number of destinations, one may collect a topology that is biased towards the vantage points.

**Load Balancing:** Another issue to keep in mind during topology collection is the deployment of load balancing by ISPs. Certain traffic engineering practices for load balancing may cause traceroute to return IP addresses that do not correspond to a real end-to-end path in the network [24]. This happens when a router forwards consecutive traceroute probes on different paths toward the destination, a common phenomenon in the Internet [25].

**Probing Overhead:** As the volume of active measurement practices has increased in time, it is important to minimize redundant probing and carefully consider any disruption that might be caused by the measurement study.

**Unresponsive Router Resolution:** Unresponsive routers are routers that are passive to measurement probes and are represented by a '*' in a traceroute output. Since a router may appear as a '*' in multiple traceroute outputs, we need to identify '*'s (i.e., unresponsive nodes) that belong to the same router[1]. This process is called as unresponsive router resolution.

[1]We use the term *unresponsive node* to refer to a '*' in a traceroute output and *unresponsive router* to refer to the actual router that is represented by this unresponsive node (i.e., by this '*') in the traceroute output.

**IP Alias Resolution:** As routers have multiple interfaces, each interface has a unique IP address. In a given set of path traces, a router may appear on multiple path traces with different IP addresses. In IP alias resolution, the goal is to identify nodes that appear to be separate in collected path traces and combine them into one single node (i.e., to detect IP addresses that belong to the same router).

**Subnet Resolution:** Normally, routers are connected to each other over subnetworks and subnet resolution helps in identifying the underlying subnets. In this task, the IP addresses in a data set are analyzed to infer subnet relations among them. The goal in subnet resolution is to identify multiple links that appear to be separate and combine them to represent their corresponding single hop connection medium (e.g., point-to-point or multi-access link).

## III. Related Work

In order to facilitate topology measurement studies, several research groups have developed mapping systems to collect the required information. Archipelago measurement infrastructure of CAIDA [26], the DIMES project [27], and the iPlane infrastructure [28] continuously provide sampled Internet topologies. Additionally, several other groups have developed tools or systems [29]–[38]. Table I presents major Internet topology mapping systems and their characteristics including number of: (1) deployed monitors, (2) destination IP addresses, (3) collected traces, (4) generated probes, (5) observed IP addresses, (6) observed edges without topology construction, (7) alias sets, (8) IP addresses that appeared in an alias set, and (9) provided data type [2]. Note that, iPlane sends a single probe per hop in collected path traces while other systems send three. Using three probes per hop helps the mapping system identify load balancing routers and carefully construct subsequent links.

*1) Ark:* Archipelago [26] is a successor of the skitter measurement infrastructure [39] that started probing the Internet in 1998. A major step from Skitter to Ark is the coordination of monitors using Marinda tuple-space, which utilizes a distributed memory space and pattern matching techniques. Ark focuses on generating annotated Internet maps. Currently, Ark utilizes 53 dedicated monitors around the world to trace every observed /24 subnetwork. Monitors are divided into 3 teams to trace towards 9.1M destination IP addresses using scamper [40] to generate approximately 100 probes per second. Ark started collecting IPv6 topology utilizing some of the monitors since September 2010.Finally, Ark utilizes Mercator, Midar, and kapar to resolve IP aliases.

*2) Dimes:* Similar to SETI@home crowd sourcing approach [41], Distributed Internet Measurements and Simulations (DIMES) [27] utilizes home computers to collect path traces around the world. Currently, around 20K agents around the world contribute as vantage points to probe destinations from a rich set of locations and capture peripheral Internet

[2]As DIMES does not release raw traces we could not obtain some of its statistics.

| Platform | Monitors | Dest. IP | Traces | Probes | IPs | Edges | Alias Sets | Aliased IPs | Data Type |
|---|---|---|---|---|---|---|---|---|---|
| Ark | 53 | 9.1M | 27.1M | 993M | 1.3M | 2.3M | 79.6K | 271K | Router/AS topology |
| DIMES | 19,000 | | | | 3.6M | 15M | | | Router/PoP/AS topology |
| iPlane | 200 | 100K | 33.8M | 472M | 0.3M | 1.2M | 12.1K | 33.2K | PoP/AS topology |
| Cheleby | 500 | 3.5M | 13.6M | 658M | 1.9M | 2.9M | 83.0K | 217K | Link-level topology |

TABLE I: Internet Topology Mapping Systems

topology. DIMES focuses on PoP level topology mapping and annotating the links with delay and loss statistics. Finally, DIMES only implements Mercator method in resolving IP aliases.

*3) iPlane:* iPlane [28] aims at providing Internet links annotated with latency, bandwidth, capacity and loss rate for improved overlay network deployment. iPlane performs path traces from 200 PlanetLab monitors towards 100K destinations to construct a backbone topology that can be used as landmarks for overlay networks. Moreover, geo-location of routers is identified using undns [15] and sarangworld [42] tools. Finally, iPlane utilizes Mercator and Ally in resolving IP aliases.

Inaccuracies in the topology sampling and construction processes may significantly affect the accuracy of the observations or results obtained in the measurement study [18]–[22], [43]. However, currently deployed topology mapping systems do not complete all topology construction tasks. In particular, Ark, DIMES and iPlane provide alias pairs for some data sets but they do not provide subnets for observed IP addresses and also ignore unresponsive routers. Addition of subnet relations and unresponsive routers in the final graph considerably improves the network accuracy.

## IV. CHELEBY: TOPOLOGY SAMPLING

In order to sample the underlying backbone topology of the Internet, Cheleby system utilizes the PlanetLab infrastructure to probe the Internet. Cheleby collects a large number of path traces from geographically diverse vantage points towards all /24 subnets in the announced subnet prefixes. Cheleby utilizes Paris traceroute, which fixes flow identifiers so that flow-identifier based load balancing routers will choose the same next hop for probe packets toward the same destination [44]. Moreover, Cheleby performs ICMP based querying as it elicits more responses than other probing approaches [45].

### *A. Destination List Generation*

In order to probe each active /24 subnetwork range, we obtain subnet announcements with originating AS from `http://www.cidr-report.org`. The list provides advertisements and actual RIR allocations for each AS. We divide each subnet advertisement into a /24 subnetwork (e.g., $A.B.C.0/24$) and pick first allocable IP address as the probing destination (i.e., $A.B.C.1$). If a specific range is smaller than /24, then we pick the first allocable IP address in the range. These IP addresses are then divided into destination blocks of approximately 1,024 destinations that will be probed by monitors. Note that, an AS might be divided into several blocks if it is larger than /14 or a destination block file may contain multiple ASes. At the end of this process, we have **3,460** destination blocks, i.e., 3.54M destination IP addresses.

After a few experiments, we replaced non-observed IP addresses with responsive IP addresses, which have a common subnetwork prefix of /24 or longer, in the earlier data sets. Moreover, we dynamically appended newly observed IP addresses to the destination lists during topology construction phase (see Section V).

### *B. Response Wait Time*

In order to determine time-out time for traceroute probes, we analyzed the response time of elicited responses for traces towards 3.54M destinations using a time-out of 1.7 seconds. Figure 2 presents the CDF of Round Trip Time (RTT) for 213.3M probes that elicited an ICMP response. In this experiment, cumulatively there were 213M responsive nodes (i.e, an ICMP response with an IP address was received) and 17.5M unresponsive nodes (i.e., no response was received) in the collected traces. In the Figure 2, we observe that more than 99.95% of responsive nodes respond within 0.5 sec. Hence, in all subsequent experiments we set time-out time to 0.5 sec since longer time-outs delay the overall topology collection process.

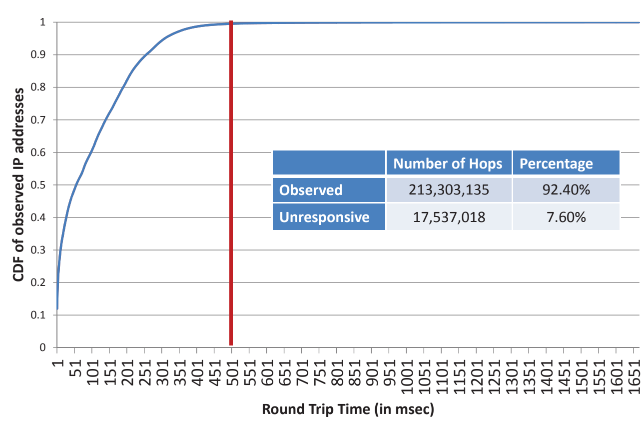



Fig. 2: Cumulative Distribution Function for RTT

### *C. Task Assignment to Monitors*

We divided functional PlanetLab nodes into 7 teams based on their geographic locations (i.e., 1: North-West America, 2: North-Central America, 3: North-East America, 4: South America, 5: Western Europe, 6: Eastern Europe + Africa + Western Asia, and 7: Eastern Asia + Australia), as shown by squares in Figure 4. The figure also shows the distribution for 5

| Team | Team 1 | Team 2 | Team 3 | Team 4 | Team 5 | Team 6 | Team 7 |
|---|---|---|---|---|---|---|---|
| Monitors | 56.63 | 53.88 | 55.50 | 56.75 | 77.25 | 73.63 | 76.25 |
| Incomplete Dest Blocks | 7.43 | 30.28 | 24.03 | 35.72 | 12.85 | 12.35 | 12.15 |
| Completed Dest Blocks | 3,453 | 3,430 | 3,436 | 3,424 | 3,447 | 3,448 | 3,448 |
| Completed in 2nd Trial | 16.2 | 63.1 | 40.6 | 60.4 | 26.9 | 23.4 | 26.1 |
| Avrg. Compl. Time (sec) | 1,476 | 1,376 | 1,586 | 1,650 | 1,764 | 1,764 | 1,566 |
| Run Time (hours) | 8.53 | 8.18 | 9.15 | 9.32 | 7.32 | 7.68 | 6.54 |

TABLE II: Team Statistics (Average of 8 Data Sets)

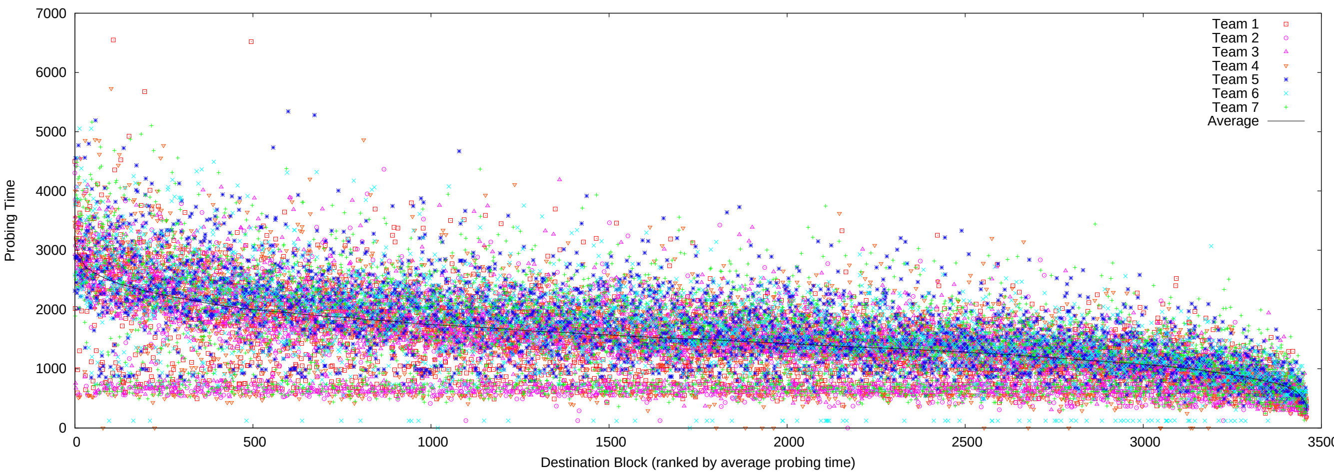



Fig. 3: Completion Time per Destination Block (in Seconds)

teams, which were deployed later as discussed in Section IV-D, with dashed lines. Ark utilizes a similar approach to divide its 53 monitors into three teams.

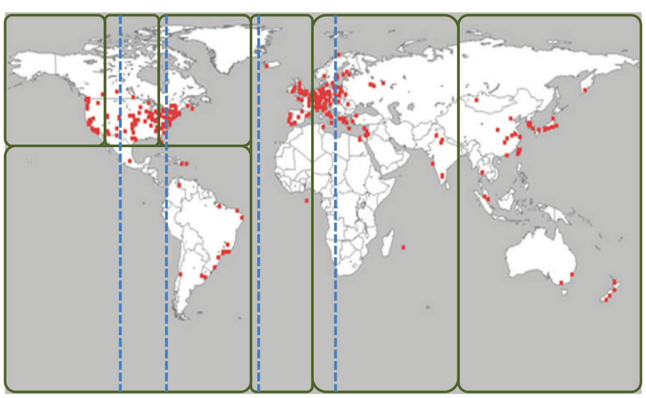


Fig. 4: Team assignment of PlanetLab nodes (5 teams: Blue lines. 7 teams: Green boxes.)

In order to probe destinations from geographically diverse vantage points, Cheleby utilizes PlanetLab [46] nodes around the world. Among ∼1,100 nodes only ∼600 of them were good to be utilized during our experiment. As ∼100 of good monitors did not function well with the Paris traceroute, we could utilize ∼500 nodes during our topology collection. In this section, we describe major steps of Cheleby regarding topology sampling and data collection experiments using ∼500 available PlanetLab nodes during Nov 2010.

Cheleby dynamically assigns one of the available monitors from each team to probe destination blocks. Each block is probed by only one monitor at a time and overall by 7 monitors (i.e., the number of teams). Each monitor is set to probe 4 destination blocks in parallel to reduce the overall round completion time. Each of the 4 monitor processes work independent from others. These processes are marked as *idle*, *busy*, or *inactive*. All processes in a monitor is *inactivated* when one of them returns its data in less than 2 minutes as this indicates a problem with the probing. They remain *inactive* for a period of 4 hours before becoming *idle* and obtaining a new destination list. Moreover, monitors are ranked based on their task completion averages and Cheleby selects the top *idle* process from a team to assign a new destination block.

Probing of a monitor is terminated if the monitor can not complete its task within a period of 2 hours. In this case, the monitor is penalized with a reduction in its ranking and brought to the *idle* state. The partially traced destination block is also put to non-probed list for another trial by another monitor in the same team. If the new monitor, which reverses the order of destination IP addresses before probing, is not able to complete probing in time as well, then the destination block is marked as partially completed and both of the partial traces are added to the database.

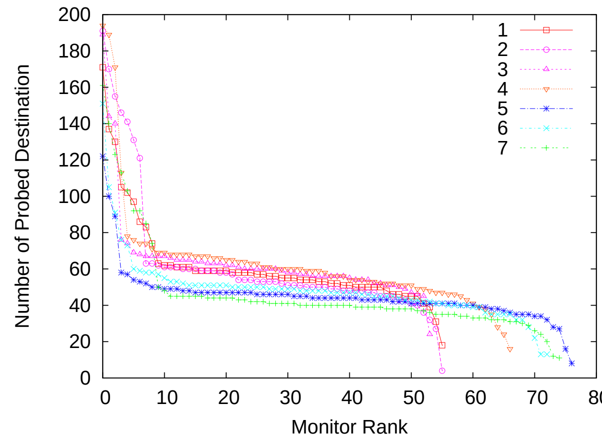
Using 7 teams, we performed 8 rounds of data collection to observe teams dynamics. Table II presents the averages of (1) the number of monitors, (2) the number of incomplete destination blocks, (3) the number of completed destination blocks,

(4) the number of destination blocks that could complete in the second trial, (5) average block completion times in seconds, and (6) total run time for each team in hours. Initially, we clustered the monitors around the world into regions to have balanced number of monitors in each team. However, teams 5, 6, and 7 were considerably behind others. Hence, we increased their monitors by adjusting geographic clusters.

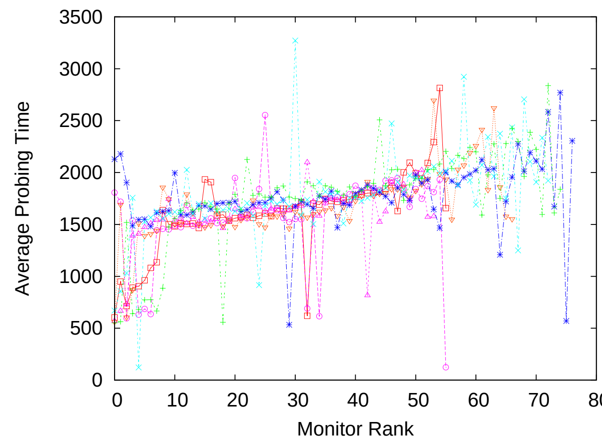
As seen in the Table II, on average 19.26 of the 3,460 destination blocks were not completed in allowed time of 2 hours even after 2nd trial. Team 4 (South America) had lowest probe completion with an average of 35.72 (i.e., 1.03% of all blocks) incomplete destination blocks. On average, 36.67 of blocks were completed in the second trial (which is included in the overall completion numbers). Team 5 (Western Europe) and Team 6 (Eastern Europe + Africa + Western Asia) were slowest with an average of 1,764 seconds to trace a destination block. This is also apparent in Figure 3, which shows destination block probing completion times of each team for a single run. However, Teams 5, 6, and 7 were the fastest ones in probing all destination blocks due to higher number of monitors in these teams. Destination blocks in the Figure 3 are ranked by the average completion times of all teams for the block from max to min (shown with black line). In Figure 3, we observe that there is a group of destinations blocks that complete probing approximately in 700 seconds independent of team averages. These cases often happen when the destination block is in the same location as the probing team.

Figure 5 displays completion time statistics for a data set. In the figures, monitors for each team are ranked by the number of destination blocks they completed probing. As seen in Figure 5-a, while most of the monitors completed 40 to 80 destinations blocks, there were outliers that either outperformed or fall behind others. Moreover, as seen in Figure 5-b, average probe completion times increased in general with lower rankings as expected. In general, the outliers that were considerably below the average curve were faulty monitors that either returned responses in few minutes, whose data was removed and set *inactive* for certain time, (e.g., Team 2 node at 55) or became available for part of the data collection (e.g., Team 5 node at 75). On the other hand, outliers well above the average line received a non-responding destination, i.e., AS regions that were not very responsive, causing jumps in completion time. Overall, the dynamic task assignment helped improve round completion time to less than half of the initial experiments where tasks were randomly assigned without timeouts and penalties.

Finally, Figure 6 presents the average of the number of unique nodes and edges observed as data from vantage points and destination blocks are respectively appended to the graph. Similar to earlier findings, we observe that addition of more monitors sub-linearly increases the number of unique IP addresses or edges. On the other hand, number of unknown nodes increases linearly as the unresponsive routers are not resolved yet and each instance is recorded as a unique node. Finally, addition of destination blocks linearly increases the




a) Average Number of Probes




b) Average Completion Time (sec)

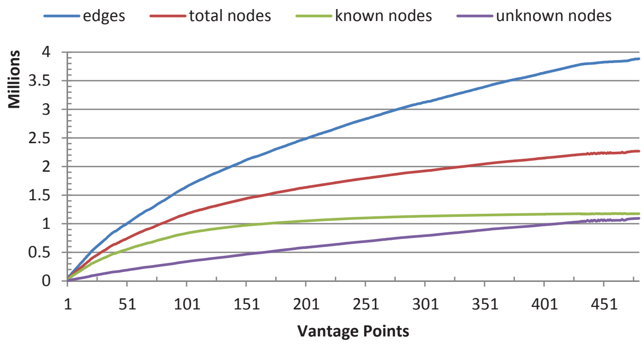
Fig. 5: Team Completion Statistics (ranked by the number of completed destination blocks)

number of observed IPs and edges as the destination blocks are for different ASes.

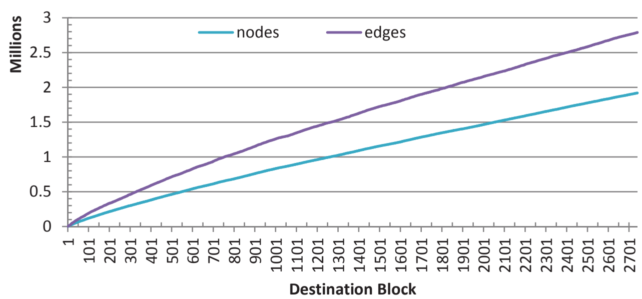
### *D. Probing Overhead Reduction*

In Cheleby, we utilize inter-monitor and intra-monitor probe reduction as shown in Figure 7. We reduce intra-monitor redundancy by performing partial traces to some destination IP addresses. Once we have a full trace to an IP address in an AS, we start successive traceroute queries from the hop distance $h_i$ of the ingress router (i.e., hop distance of the last IP address in the trace that did not belonging to the destination AS). If the first IP of the new trace has not appeared at the same hop distance $h_j$ in any of the earlier full traces to the AS, then we complete the trace. Otherwise, we do not complete the trace. Analyzing collected traces we observe that 35.4 % of 22.4M traces are partial traces. This overall saved 66.2M probes that would be generated with full tracing.

Additionally, to reduce inter-monitor redundant probing, a destination IP is probed by only one monitor of a team. Since the monitors in the same team are geographically close to each other, we expect their contribution to identify a new link/node




a) Cumulative Monitor Nodes and Edges




a) Cumulative Destination Block Known nodes and Edges

Fig. 6: Number of Nodes and Edges (average of 8 data sets)

| Teams | 3 | 5 | 7 | 9 | 11 |
|---|---|---|---|---|---|
| Time (min) | 540 | 630 | 770 | 1,220 | 1,540 |
| Traces | 9.5M | 15.9M | 22.0M | 28.7M | 35.0M |
| Probes | 151M | 249M | 347M | 452M | 552M |
| Total IPs | 95.3M | 157M | 219M | 285M | 348M |
| Total *s | 55.7M | 92.4M | 128M | 167M | 204M |
| Unique IPs | 1.11M | 1.18M | 1.21M | 1.24M | 1.27M |
| IPs / all | 79.3% | 84.3% | 86.3% | 88.8% | 90.7% |
| Per min IPs | 2,057 | 1,874 | 1,571 | 1,020 | 825 |
| Unique Edges | 1.42M | 1.76M | 1.96M | 2.13M | 2.26M |
| Edges / all | 46.1% | 57.1% | 63.6% | 69.1% | 73.1% |
| Per min Edges | 2,636 | 2,794 | 2,550 | 1,747 | 1,465 |

TABLE III: Team Statistics with Different Team Sizes

is small. Moreover, we are in the process of identifying ingress points of ASes to dynamically establish teams for each destination AS so that we have exactly one monitor probing through each ingress point of an AS. That is, we will determine the sets of monitors that probe each ingress point of the AS and then build individual teams for each AS IP addresses.

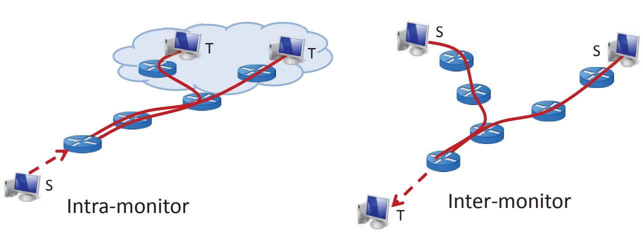
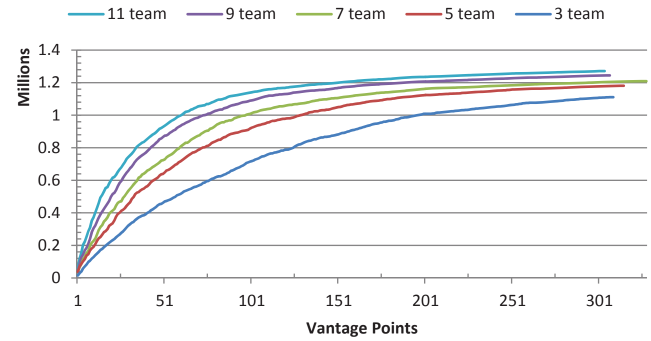
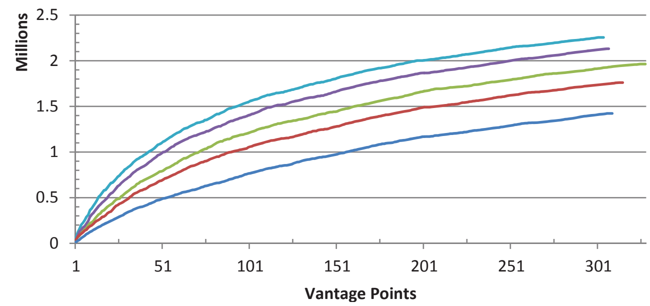



Fig. 7: Intra- and Inter-monitor Redundancy Reduction




a) Number of Known Nodes




b) Number of Edges

Fig. 8: Number of Nodes and Edges for different team sizes

We performed an experiment where we varied team sizes to analyze the effect of choosing different number of teams. Variations in the number of teams has a direct effect as seen in the Table III, which for each team configuration presents: (1) the round completion time, (2) generated traces, (3) generated probes, (4) probes yielding an IP address, (5) probes that did not elicit a response, (6) unique IP addresses, (7) percentage of observed IP addresses compared to combination of all IP addresses, (8) number of observed IP addresses per minute of probing, (9) unique edges, (10) percentage of observed edges compared to union of all, and (11) number of observed edges per minute of probing. Additionally, Figure 8 presents the changes in the number of observed IP addresses and edges with aggregation of monitor data.

As the number of teams increases more probes are generated and less monitors are deployed per team. Both of these cause longer round completion times. However, as seen in unique IPs and unique edges rows, there is a diminishing benefit with higher number of probes. Even though, using 11 teams returns highest number of unique IPs and unique edges, the overhead is highest per observed IP address. An important observation is that the overlap between the edges is much smaller than the overlap between the IP addresses because deployed monitors in each case differ. Considering this analysis, we decided to utilize 7 teams in the Cheleby as it provides best balance between coverage and overhead.

## V. Cheleby: Topology Construction

After collecting topology data, we need to process this raw data to obtain the underlying network topology. In particular, we (1) filter faulty traces, i.e., initial pruning, (2) infer underlying physical subnets among IP addresses, (3) resolve IP addresses belonging to the same router, and (4) resolve unresponsive routers as shown in Figure 9. These resolution tasks especially are challenging when large scale topologies of millions of nodes are processed. In this section, we analyze each of these tasks and indicate the algorithms that we utilized to handle these tasks.

### *A. Initial Pruning*

As path traces contain anomalies such as routing loops, we first prune raw path traces. The pruning breaks path traces with a loop (e.g., $IP_A$, $IP_B$, $IP_C$, $IP_D$, $IP_E$, $IP_C$, $IP_F$, $IP_G$) into three pieces based on the repeated IP address (i.e., $IP_C$) and utilize the first part (i.e., $IP_A$, $IP_B$, $IP_C$) and the last part (i.e., $IP_C$, $IP_F$, $IP_G$) of the trace in the remainder of processing. In data sets collected with 7 teams, 772K (%3.45 of 22.4M) of path traces contain routing loops among which 143K has multiple loops. Moreover, we observed border firewalls that filter ICMP packets from/to a network domain and occasionally respond with their IP address. However, the hop distance of these IP addresses are not consistent. Hence, we filter any IP address that appears at the end of a trace after three anonymous nodes.

We build initial network graph by parsing filtered path traces. During parsing, we resolve unknown nodes that are between the same set of known nodes by detecting the same *-substrings (i.e., the same length *-substrings with the same known nodes as the end points). Performing this unresponsive router resolution step during graph construction reduces the number of unknown nodes by %78.71 on average. Table IV presents the number of (1) all traces, (2) partial traces, (3) saved probes, (4) unknown nodes, i.e., '*', and (5) known nodes, i.e., IP addresses, for the analyzed data sets.

### *B. Subnet Inference*

First task after building an initial network graph is the identification of the underlying physical subnets, i.e., link level connectivity, among IP addresses in the collected topology [16]. The goal in subnet resolution is to identify multiple links that appear to be separate and combine them to represent their corresponding single hop connection medium (i.e., multi-access link). Subnet resolution also finds the missing links between IP addresses that fall in the same subnet range but were not observed in path traces. The successful inclusion of subnet relations among the routers yields topology maps that are closer, at the subnet level, to the sampled segments of the Internet.

Cheleby, enhances subnet resolution approach presented in [16] by utilizing only the distance preservation condition but not the trace preservation condition to reduce the computational complexity. SubNet Inferrer module (SNI) observes distances of all IP addresses per vantage point and determines IP address ranges that have similar distances to all vantage points. Different from initial approach in [16], we only allow one IP address being closer to each of the vantage points.

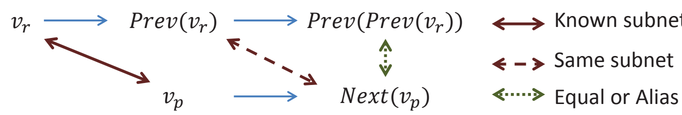



Fig. 10: Analytical and Probe-based Alias Resolver v2

As the number of vantage points is increased, the distance condition can more accurately filter false subnets without relying on the trace accuracy condition.

Table V presents averages of identified subnets and the completenesses of the subnets that had %20 of their IP addresses present in the trace data set. This number is less than we expected as only 99K of collected 1.2M IP addresses appear in a subnet. The main reason for this is because we did not explore other IP addresses of candidate subnets. We then add a probing module into SNI that probes subnets that have less than half of their IP addresses present in the data set as explained in Section V-E.

### *C. IP Alias Resolution*

After inferring underlying subnets, Cheleby resolves IP aliases using *Analytic and Probe-based Alias Resolver (APAR)* [47]. As pointed by Keys in [48], original APAR implementation had high storage requirements[3]. Similar to kapar, we enhanced APAR implementation by eliminating path queries (called APARv2) as shown in Figure 10. During APAR neighbor matching (see [47] for details), we need to verify whether our candidate alias pair (i.e., $v_p$ and $Prev(v_r)$) has a common neighbor (i.e., $Prev(Prev(v_r))$ and $Next(v_p)$) as an alias or as in another subnet relation (i.e., $Prev(v_r)$ and $Next(v_p)$). Hence, for each node in the graph, we record previous nodes and next nodes from path traces, and derive 2-hop predecessors of the IP addresses. We also record conflict sets, i.e., set of traces an IP address appeared in, to ensure trace accuracy condition. These changes help us eliminate the need to keep path traces in memory and query them during alias resolution.

Utilizing APARv2 on collected data, we identified 23,266 alias sets that include 75,019 aliased IP addresses on average. However, this corresponds to only $sim$7% of observed IP addresses. This value was especially low as we did not include IP-mates (/30 or /31 pair of the observed IP address) as in Ark and iPlane and we had a low subnet coverage as these subnets help in alias identification. Hence, we (1) improve subnet coverage with probing candidate subnet IP addresses, (2) include IP-mate probing component into APARv2, and (3) implemented probing based mercator and ally appraches to complement APARv2 as described in Section V-E.

### *D. Unresponsive Router Resolution*

Unresponsive routers are routers that are passive to measurement probes and are represented by a '*' in a traceroute output. In Cheleby, we utilize our *Graph Based Induction* (GBI)

[3]Note that, other improvements proposed by [48] were discussed in [47] and presented as options.

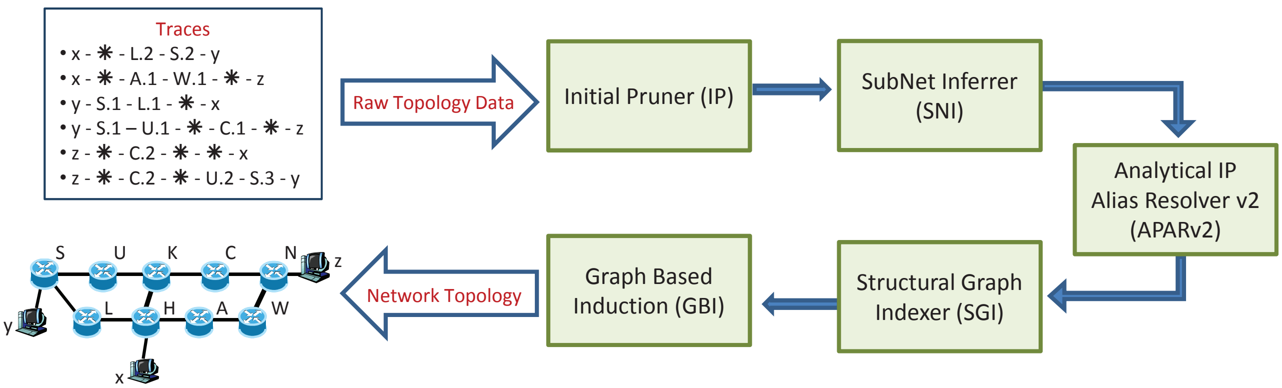



Fig. 9: Topology Construction

| Data Set | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| All Traces | 22.39 | 22.42 | 22.42 | 22.40 | 22.42 | 22.41 | 22.42 | 22.03 |
| Partial Traces | 8.02 | 8.12 | 8.05 | 7.86 | 7.67 | 7.98 | 7.91 | 7.80 |
| Saved Probes | 65.23 | 66.14 | 67.68 | 66.32 | 63.98 | 67.90 | 66.19 | 65.98 |
| Unknown Nodes | 4.93 | 4.81 | 4.90 | 4.88 | 4.95 | 4.94 | 4.95 | 4.92 |
| Known Nodes | 1.18 | 1.18 | 1.19 | 1.17 | 1.20 | 1.17 | 1.19 | 1.17 |

TABLE IV: Topology Data (in millions)

| Subnet Size | /24 | /25 | /26 | /27 | /28 | /29 | /30 | /31 |
|---|---|---|---|---|---|---|---|---|
| Count | 0.38 | 4.25 | 34.13 | 485 | 6,381 | 20,602 | 11,202 | 2,960 |
| Completeness | 27.7% | 24.5% | 23.3% | 23.3% | 24.8% | 36.0% | 100% | 100% |
| All IPs | 26 | 131 | 492 | 3,383 | 22,110 | 44,500 | 22,403 | 5,920 |

TABLE V: Average Subnet Statistics for 8 Data Sets

| Initial | I. Pruner | Rate Lim. | Triangle | Bipartite | Star | Final *s |
|---|---|---|---|---|---|---|
| 7,207,885 | 6,137,750 | 51,279 | 2,858 | 143,880 | 619,204 | 252,915 |

TABLE VI: Unresponsive Router Resolution (Average of 8 data sets)

technique to resolve unresponsive routers [49]. We enhanced GBI using our structural graph indexer (SGI) [50], which helps improve subsequent graph queries in the graph database, to reduce the search time of GBI. SGI indexes maximal graphs that match the structure formulation within the original graph in a consecutive manner. SGI first identifies star structures, then complete-bipartite, triangle and finally clique structures from the preceding ones. In our experiments, we realized that the number of cliques with more than three nodes is minimal and hence we removed clique indexing from Cheleby. After indexing structures with SGI, Cheleby resolves corresponding unresponsive routers using GBI obeying the trace preservation condition.

Table VI presents averages of unresponsive router resolution steps. As indicated in Section V-A, initial pruning resolves considerable number of unknown nodes. Then using SGI, we perform GBI on remaining ones to reduce the number of final unresponsive routers to 250K. This yields topologies where 17.24% of the routers are unresponsive, which agrees with our earlier observations [45].

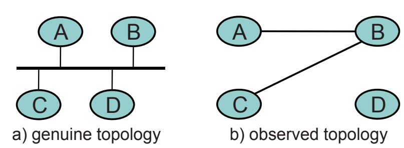



Fig. 11: Effect of subnet resolution

### E. Increasing Graph Density

Realizing that many subnets had low completeness, we decided to increase the coverage as indicated in [16]. For this, we determined non-observed IP addresses of subnets that had at least 10% completeness. For the last data-set there were 651.8K IP addresses missing from the identified candidate subnets. Moroever, we looked at /30 and /31 mate of observed IP addresses and they produced 535.2K and 93.1K IP addresses, respectively. Next, to ensure the existence of these IP addresses, we performed a reverse DNS lookup and probed them with a ping. If either of these tests were positive, we added them to the destination IP lists.

| Subnet Size | /24 | /25 | /26 | /27 | /28 | /29 | /30 | /31 |
|---|---|---|---|---|---|---|---|---|
| Count | 4 | 36 | 184 | 1,294 | 8,836 | 93,110 | 20,543 | 37,468 |
| Completeness | 26.3% | 30.0% | 28.3% | 27.7% | 28.0% | 39.3% | 100% | 100% |
| All IPs | 268 | 1,359 | 3,228 | 10,767 | 34,587 | 219,745 | 41,086 | 74,936 |

TABLE VII: Improved Subnet Statistics

| Resolver | Alias Sets | Aliased IPs |
|---|---|---|
| APARv2 | 38,012 | 128,495 |
| Ally (path traces) | 32,860 | 65,720 |
| Ally (common neighbor) | 32,595 | 65,190 |
| Ally (subnet) | 25,436 | 50,872 |
| Ally (combined) | 55,027 | 110,054 |
| Mercator | 305 | 610 |
| Combined | 82,962 | 216,628 |

TABLE VIII: Alias Resolution Statistics

After these changes, we obtain a better resolution and a more complete topology. As seen in table VII, the number of observed subnets and their completeness significantly increased. In final topology, the number of IP addresses observed in a subnet is about 400K, which is four times of the initial 99K.

Moreover, improvements in the subnet coverage and inclusion of IP-mates considerably improved alias IPs identified with APARv2 as seen in Table VIII. The number of alias sets increased from 23K to 38K and the number of IP addresses in an alias set increased from 75K to 128K. Additionally, we implemented probing based mercator and ally approaches to complement APARv2. For mercator, we sent a probe to all observed IP addresses and recorded the response. If the response was from an IP address different from queried one, then we marked them as aliases. This approach produced the least number of aliases, i.e. only 610 IP addresses were placed in an alias set.

Moreover, we utilized ally on candidate alias IP address pairs. For this, we identified candidates using three methods. First, we identified path traces that had multiple IP addresses at a given hop distance. Then, we marked 70K IP address pairs at the same hop as candidate aliases to be probed with ally. Next, in the final graph, we identified IP addresses that had the same common neighbors, i.e., IP addresses whose neighbor intersection was more than one node. Similarly, we marked 2M pairs of these IP addresses as candidates. Finally, we used subnets as pivot points to determine candidate aliases. For each subnet (e.g., consider subnet in Figure 11-a), we marked each subnet IP address (e.g., A, B, C and D) with other the IP addresses' neighboring IP addresses (i.e., A with the neighbors of {B, C, D}; B with the neighbors of {A, C, D}; C with the neighbors of {A, B, D}; and D with the neighbors of {A, B, C}). Subnet based candidate generation produced 3M alias pairs. Probing these pairs with ally we identified aliases for 66K, 65K, and 51K IP addresses, respectively.

After combining all these resolved alias sets, we obtain 83K alias sets that contain 217K IP addresses, which is more than three times of the initial resolution results.

## VI. Conclusion

Due to the tremendous growth in Internet's importance, many groups, organizations, and governments have become interested in understanding various characteristics of the Internet for commercial, social, and technical reasons. Network research community depends on such Internet mapping systems to understand characteristics of the Internet and develop new protocols and services. Government agencies are interested in Internet measurements to protect and improve the national cyber infrastructure. Moreover, new network paradigms such as overlay networks require knowledge of the underlying network topology.

In this paper, we presented Cheleby Internet topology mapping system that provides sample network topologies at the subnet level (available at `http://cheleby.cse.unr.edu`). Cheleby is an assembly of topology collection and construction techniques, i.e., target list generation, probing redundancy reduction, unbiased accurate data collection, subnet inference, alias resolution, and unresponsive router resolution, into a single system. Note that, the validity of all of these approaches are discussed in the respective papers in greater detail. Moreover, the lack of public knowledge about large-scale Internet topologies necessitates the Internet topology measurement studies.

Cheleby system improves earlier systems by incorporating topology construction steps in produced topology data. More specifically, neither of existing systems provide subnetworks for collected IP addresses and they discard anonymous nodes in the final topologies. These processes may have considerable effect on observed network characteristics.

## References


[1] R. Pastor-Satorras and A. Vespignani, *Evolution and Structure of the Internet*. Cambridge University Press, 2004.

[2] M. Crovella and B. Krishnamurthy, *Internet Measurement: Infrastructure, Traffic and Applications*. Wiley, 2006.

[3] H. Haddadi, M. Rio, G. Iannaccone, A. Moore, and R. Mortier, "Network topologies: inference, modeling, and generation," *Communications Surveys Tutorials, IEEE*, vol. 10, no. 2, pp. 48 –69, second 2008.

[4] L. Cheng, N. Hutchinson, and M. Ito, "Realnet: A topology generator based on real Internet topology," in *Advanced Information Networking and Applications - Workshops, 2008. AINAW 2008. 22nd International Conference on*, 25-28 2008, pp. 526 –532.

[5] D. Johnson, D. Gebhardt, and J. Lepreau, "Towards a high quality path-oriented network measurement and storage system," in *PAM'08: Proceedings of the 9th international conference on Passive and active network measurement*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 102–111.

[6] E. Katz-Bassett, H. Madhyastha, J. John, A. Krishnamurthy, D. Wetherall, and T. Anderson, "Studying black holes in the Internet with hubble," in *Proceedings of USENIX Symposium on Networked Systems Design and Implementation*, San Fransicso, CA, USA, April 2008.

[7] K. Naidu, D. Panigrahi, and R. Rastogi, “Detecting anomalies using end-to-end path measurements,” in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, 13-18 2008, pp. 1849 –1857.
[8] Y. Zhao, Y. Chen, and D. Bindel, “Towards unbiased end-to-end network diagnosis,” *Networking, IEEE/ACM Transactions on*, vol. 17, no. 6, pp. 1724 –1737, dec. 2009.
[9] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, “The flattening Internet topology: natural evolution, unsightly barnacles or contrived collapse?” in *PAM’08: Proceedings of the 9th international conference on Passive and active network measurement*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 1–10.
[10] H. Haddadi, D. Fay, S. Uhlig, A. Moore, R. Mortier, and A. Jamakovic, “Mixing biases: Structural changes in the AS topology evolution,” in *TMA ’10: Proceedings of the Second International Workshop on Traffic Measurements and Analysis*. Springer, April 2010.
[11] Y. He, G. Siganos, M. Faloutsos, and S. Krishnamurthy, “Lord of the links: A framework for discovering missing links in the Internet topology,” *Networking, IEEE/ACM Transactions on*, vol. 17, no. 2, pp. 391 –404, april 2009.
[12] P. Mahadevan, D. Krioukov, M. Fomenkov, X. Dimitropoulos, k. c. claffy, and A. Vahdat, “The Internet AS-level topology: three data sources and one definitive metric,” *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, pp. 17–26, 2006.
[13] D. Fay, H. Haddadi, A. Thomason, A. Moore, R. Mortier, A. Jamakovic, S. Uhlig, and M. Rio, “Weighted spectral distribution for Internet topology analysis: Theory and applications,” *Networking, IEEE/ACM Transactions on*, vol. 18, no. 1, pp. 164 –176, feb. 2010.
[14] K. Yoshida, Y. Kikuchi, M. Yamamoto, Y. Fujii, K. Nagami, I. Nakagawa, and H. Esaki, “Inferring pop-level ISP topology through end-to-end delay measurement,” in *PAM ’09: Proceedings of the 10th International Conference on Passive and Active Network Measurement*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 35–44.
[15] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, “Measuring ISP topologies using rocketfuel,” *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, pp. 2–16, February 2004.
[16] M. Gunes and K. Sarac, “Inferring subnets in router-level topology collection studies,” in *ACM SIGCOMM Internet Measurement Conference (IMC)*, San Diego, CA, October 2007.
[17] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson, “Inferring link weights using end-to-end measurements,” in *IMW ’02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*. New York, NY, USA: ACM, 2002, pp. 231–236.
[18] L. Amini, A. Shaikh, and H. Schulzrinne, “Issues with inferring Internet topological attributes,” in *Proceedings of SPIE ITCom*, Boston, MA, USA, July/August 2002.
[19] M. Gunes, S. Bilir, K. Sarac, and T. Korkmaz, “A measurement study on overhead distribution of value-added Internet services,” *Computer Networks*, vol. 51, no. 14, pp. 4153–4173, 2007.
[20] A. Lakhina, J. Byers, M. Crovella, and P. Xie, “Sampling biases in IP topology measurements,” in *Proceedings of IEEE INFOCOM*, San Francisco, CA, USA, March/April 2003.
[21] R. Teixeira, K. Marzullo, S. Savage, and G. Voelker, “In search of path diversity in ISP networks,” in *Proceedings of the USENIX/ACM Internet Measurement Conference*, Miami, FL, USA, October 2003.
[22] S. Bilir, K. Sarac, and T. Korkmaz, “Intersection characteristics of end-to-end Internet paths and trees,” in *IEEE International Conference on Network Protocols (ICNP)*, Boston, MA, USA, November 2005.
[23] S. Kandula and R. Mahajan, “Sampling biases in network path measurements and what to do about it,” in *IMC ’09: Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*. New York, NY, USA: ACM, 2009, pp. 156–169.
[24] B. Augustin, F. Friedman, and R. Teixeira, “Measuring loadbalanced paths in the Internet,” in *Proceedings of IMC*, San Diego, CA, October 2007.
[25] M. Antic and A. Smiljanic, “Routing with load balancing: increasing the guaranteed node traffics,” *Communications Letters, IEEE*, vol. 13, no. 6, pp. 450 –452, june 2009.
[26] K. Claffy, Y. Hyun, K. Keys, M. Fomenkov, and D. Krioukov, “Internet mapping: From art to science,” *Conference For Homeland Security, Cybersecurity Applications & Technology*, vol. 0, pp. 205–211, 2009.
[27] Y. Shavitt and E. Shir, “DIMES: Let the Internet measure itself,” *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 5, pp. 71–74, 2005.
[28] H. V. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, and A. Venkataramani, “iPlane: An information plane for distributed services,” in *OSDI*, November 2006.
[29] B. Donnet, P. Raoult, T. Friedman, and M. Crovella, “Efficient algorithms for large-scale topology discovery,” in *Proceedings of ACM/SIGMETRICS*. New York, NY, USA: ACM Press, 2005, pp. 327–338.
[30] T. McGregor, S. Alcock, and D. Karrenberg, “The RIPE NCC Internet Measurement Data Repository,” in *Proceedings of the 11th Passive and Active Measurement Conference (PAM 2010)*, April 2010. [Online]. Available: http://www.pam2010.ethz.ch/papers/full-length/12.pdf
[31] R. Govindan and H. Tangmunarunkit, “Heuristics for Internet map discovery,” in *IEEE INFOCOM*, Tel Aviv, ISRAEL, March 2000.
[32] S. Kim and K. Harfoush, “Efficient estimation of more detailed Internet IP maps,” in *IEEE International Conference on Communications (ICC)*, Glasgow, Scotland, June 2007.
[33] Z. Wen, S. Triukose, and M. Rabinovich, “Facilitating focused Internet measurements,” in *SIGMETRICS ’07: Proceedings of the 2007 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. New York, NY, USA: ACM, 2007, pp. 49–60.
[34] University of Oregon, “Route views project,” http://www.routeviews.org.
[35] N. Spring, D. Wetherall, and T. Anderson, “Scriptroute: A public Internet measurement facility,” in *North American Network Operator’s Group*, October 2002. [Online]. Available: citeseer.ist.psu.edu/spring02scriptroute.html
[36] A. McGregor, H.-W. Braun, and J. Brown, “The NLANR network analysis infrastructure,” *IEEE Communications Magazine*, vol. 38, no. 5, pp. 122–128, May 2000.
[37] V. Paxson, J. Mahdavi, A. Adams, and M. Mathis, “An architecture for large-scale Internet measurement,” *IEEE Communications*, vol. 36, no. 8, pp. 48–54, August 1998. [Online]. Available: citeseer.ist.psu.edu/paxson98architecture.html
[38] N. Spring, D. Wetherall, and T. Anderson, “Reverse engineering the Internet,” *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 1, pp. 3–8, 2004.
[39] D. McRobb, K. Claffy, and T. Monk, *Skitter: CAIDA’s macroscopic Internet topology discovery and tracking tool*, 1999, available from http://www.caida.org/tools/skitter/.
[40] *Scamper*, http://www.wand.net.nz/scamper/.
[41] D. P. Anderson, J. Cobb, E. Korpela, M. Lebofsky, and D. Werthimer, “Seti@home: an experiment in public-resource computing,” *Commun. ACM*, vol. 45, no. 11, pp. 56–61, 2002.
[42] *Sarangworld project*, http://www.sarangworld.com/TRACEROUTE/.
[43] M. Gunes and K. Sarac, “Importance of IP alias resolution in sampling Internet topologies,” in *IEEE Global Internet (GI)*, Anchorage, AK, May 2007.
[44] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira, “Avoiding traceroute anomalies with Paris traceroute,” in *Proceedings of IMC*, Rio de Janeiro, Brazil, October 2006.
[45] M. H. Gunes and K. Sarac, “Analyzing router responsiveness to active measurement probes,” in *PAM ’09: Proceedings of the 10th International Conference on Passive and Active Network Measurement*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 23–32.
[46] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman, “PlanetLab: an overlay testbed for broad-coverage services,” *SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 3, pp. 3–12, 2003.
[47] M. Gunes and K. Sarac, “Resolving IP aliases in building traceroute-based Internet maps,” *Networking, IEEE/ACM Transactions on*, vol. 17, no. 6, pp. 1738 –1751, dec. 2009.
[48] K. Keys, “Internet-scale IP alias resolution techniques,” *SIGCOMM Comput. Commun. Rev.*, vol. 40, pp. 50–55, January 2010. [Online]. Available: http://doi.acm.org/10.1145/1672308.1672318
[49] M. H. Gunes and K. Sarac, “Resolving anonymous routers in Internet topology measurement studies,” in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, 13-18 2008, pp. 1076 –1084.
[50] H. Kardes and M. H. Gunes, “Structural graph indexing for mining complex networks,” in *ICDCS Workshop on Simplifying Complex Networks for Practitioner (SIMPLEX 2010)*, June 2010.