Identifying and Aggregating Homogeneous IPv4 /24 Blocks with Hobbit

Youndo Lee
University of Maryland
ydlee@cs.umd.edu

Neil Spring
University of Maryland
nspring@cs.umd.edu

ABSTRACT

Addresses in the Internet are typically measured as if they represent larger aggregates. These larger blocks may be based on prefixes advertised through BGP, with larger prefixes broken into "/24s." Such an approach is typical in network mapping and other research, and tries to balance the detail available by probing more addresses with the efficiency available by probing only as many as will discover new information.

In this paper, we consider prefix homogeneity: the extent to which addresses within the same prefix are co-located in topology and have similar performance. We consider whether "24" is the right unit of homogeneity, whether additional efficiency is possible by using larger or even discontiguous address aggregates in some cases, and in what situations additional detail may be missed by treating addresses as representative of /24 blocks. With these results, we present a map of homogeneous address aggregates in the network.

Keywords

Topological proximity; IPv4 /24 block; Last-hop router

1 INTRODUCTION

IPv4 addresses are commonly represented in dot decimal notation where each of four octets is written in decimal numbers and concatenated with dots. In this notation, it is very straightforward to discern whether addresses are in the same /24. Although it is not clear whether this notation promoted the use of /24 prefixes or vice versa, it is true that /24 is a very common prefix. For example, 53% of BGP prefixes (obtained from RouteViews BGP snapshot) are /24 prefixes.

The wide use of /24 prefixes can be a good reason for considering a /24 block (that is, a block consisting of IPv4 addresses having a common /24 prefix) as a unit. Actually, several measurement studies and systems use /24 blocks as a unit mostly for the purpose of reducing measurement loads, possibly at the expense of accuracy or completeness. An Internet outage detection system called Trinocular [1] tracks outages for /24 blocks, and a recent study on the availability of Internet hosts have focused on the availability of /24 blocks [2]. The IPv4 topology dataset of CAIDA [3] is constructed by probing the destinations randomly chosen from each routed /24 prefix. The EDNS-Client-Subnet DNS extension [4] strongly encourages recursive resolvers to truncate the IPv4 addresses of users to 24 bits, for the purpose of protecting the privacy of users.

The performance of the systems using /24 blocks as a unit is closely related to the homogeneity of /24 blocks, because a lack of homogeneity may hurt their operations. For example, Trinocular may fail to detect outages if a few addresses within a /24 block have an outage while others are normally up. The EDNS-Client-Subnet extension may also fail to find the single best server for addresses within a /24 block if some addresses are distant from each other.

In this paper, we measure homogeneity of /24 blocks to verify whether /24 is a good unit. We focus on topological proximity because it is closely related to the operations of the most of the systems that use /24 blocks as a unit. If the addresses within /24s are topologically distant, they are unlikely to have identical traceroute results (thus affecting topology discovery by CAIDA), concurrent outages (affecting Trinocular) and identical corresponding front-end servers (affecting the EDNS extension).

The measurement of homogeneity in terms of topological proximity may seem to be a trivial problem which can be simply solved by using traceroute. However, due to the prevalence of load-balancing, comparing the traceroute results is not straightforward. While Paris-traceroute deals with load-balancing, it is not a panacea, particularly for per-destination load-balancing. A challenge is that per-destination load-balancing is prevalent and it often changes even the last-hop routers of the topologically co-located addresses. To address this

IMC 2016, November 14 - 16, 2016, Santa Monica, CA, USA

 ISBN 978-1-4503-4526-2/16/11…$15.00

DOI: http://dx.doi.org/10.1145/2987443.2987448

challenge, we develop a methodology that distinguishes differences in the routes due to load-balancing from those due to distinct route entries. Using this methodology, which we call *homogeneous block identification technique* (Hobbit), we evaluate the homogeneity of /24 blocks.

Even if /24 blocks are homogeneous, they may not become a good unit if they can be further aggregated into larger homogeneous blocks. Therefore, our work in this paper includes finding homogeneous blocks larger than /24 blocks by aggregating homogeneous /24s. Given a list of homogeneous /24 blocks, we first associate each /24 with its topology information. An obvious way of the aggregation is to merge /24 blocks that have exactly the same information. We first aggregate the blocks in this way, and then try to merge the /24 blocks with similar but not identical information. Even if /24 blocks are topologically co-located, their topology information gathered by our methodology may not be identical (e.g., due to a relatively small number of information sources, that is, responsive addresses within some /24s compared to the others). We deal with this using a graph clustering algorithm called MCL [5]. We represent /24 blocks as vertices and the similarities of the topology information between /24s as edges, and then apply MCL.

Our work of finding homogeneous blocks has several implications. First, homogeneous blocks larger than /24 can be used for improving efficiency of the systems using /24 blocks as a unit. For example, by choosing destinations for topology discovery from each of the identified homogeneous blocks which could be larger than /24 (instead of from each /24), measurement loads can be reduced. The saved resources can be utilized for enhancing the coverage of the topology dataset by sending more probes to the heterogeneous /24 blocks we find. Second, the identification of homogeneous blocks enables stratified sampling. Compared to simple random sampling, stratified sampling is more likely to choose representative samples because it draws samples from each homogeneous sub-group. Since IPv4 addresses are owned by diverse organizations, the advantage of stratified sampling can be significant. Finally, homogeneous blocks can provide guidance in searching for new addresses of the hosts that changed their addresses by DHCP. To characterize the behaviors of specific hosts, it may not be enough to identify their addresses once and keep tracking the addresses because of dynamic addresses. If there is no way of new addresses being informed by the hosts, the new addresses need to be searched for. Knowing the addresses that are in the same homogeneous blocks as their (old) addresses can help this search.

We make the following contributions in this paper.

- We develop a methodology called Hobbit that distinguishes between route differences due to load-balancing and distinct route entries.
- We find 1.77M homogeneous /24 blocks using Hobbit (which accounts for 90% of the /24s that were measurable).
- We analyze the composition of heterogeneous /24 blocks.
- We find 131k homogeneous blocks larger than /24s, whose size ranges from 2 to 1,251 in terms of the /24s they contain.
- We characterize top 15 biggest homogeneous blocks.

The remainder of the paper is structured as follows. We present the basic idea of a methodology for measuring homogeneity of /24 blocks in Section 2. We detail our methodology in Section 3, and describe measurement results in Section 4. Section 5 and 6 describe how to aggregate homogeneous /24 blocks and present the aggregation results. We discuss implications of our work in Section 7. We compare our work to related work in Section 8 and conclude in Section 9 with future work.

2 METHODOLOGY

A straw-man proposal for measuring the homogeneity of /24 blocks is to obtain IP-level routes of all the addresses within /24 and conclude that a /24 is homogeneous if all the IP-level routes are identical. An underlying assumption of this approach is that the routes towards co-located addresses are identical. However, in today's Internet where path diversity due to load-balancing is prevalent, this is not true for many addresses. Even probes between the same source-destination pairs often take different paths [6]. We first describe how to deal with load-balanced paths.

2.1 Paris-traceroute is helpful but not enough

Paris-traceroute, which is a variant of traceroute, has been proposed to correct inaccurate inferences of paths due to load-balancing. It tunes the values of the packet header fields that affect the path selection by load-balancers, so that all probes towards a destination follow the same path. Paris-traceroute can also be extended to a tool[1] that enumerates all paths between a source-destination pair.

We use Paris-traceroute MDA in comparing (IP-level) routes of different addresses, to prevent from falsely classifying identical routes as being different. If the numbers of routes towards destinations are more than one, identifying a single route for each destination may cause false classifications. For example, if destinations A and B both have routes $\{r_1, r_2\}$, and we find only a single route r_1 for A and r_2 for B, then A and B will appear to have different routes which is not true. To prevent this from happening, we enumerate all routes using Paris-traceroute MDA and compare the sets of routes.

[1]The extended version is called Multipath detection algorithm (MDA). In this paper, we use the term "Paris-traceroute MDA" because MDA is often considered as a subcomponent of Paris-traceroute.

Based on the methodology described above, we perform a preliminary analysis on the homogeneity of /24 blocks. We first identify active IPv4 addresses using ZMap ICMP Echo Request scan dataset [7, 8]. This dataset is generated by sending ICMP Echo request probes to all public IPv4 addresses, and recording the reply messages (if exist). We only consider IPv4 addresses that responded with ICMP Echo reply messages to be active[2]. Given the list of active addresses, we select an active address from each /26 block while excluding /24 blocks that have no active address in any of the /26s within them. In other words, we only select /24s that have at least one active address in every /26 block within them, to increase the confidence of our result to represent /24s not /25s nor /26s. For each chosen address, we enumerate all the routes between a source located at UMD and the chosen address. We consider that addresses have identical routes if they share at least one route. A /24 block is regarded as being homogeneous if all of the (four) addresses within the block have identical routes. To our surprise, 88% of the /24 blocks were *heterogeneous*. Considering that we address per-flow load-balancing using Paris-traceroute MDA and that we are generous in determining whether addresses have identical routes by requiring only one route to be identical, 88% is unexpectedly high. With a doubt that ICMP rate limiting can be a confounding factor, we try to eliminate the effect of ICMP rate limiting. We use unresponsive hops as wildcards that can represent any address in comparing routes. For example, routes <A.A.A.A, B.B.B.B, C.C.C.C>, <A.A.A.A, *, C.C.C.C> and <*, B.B.B.B, C.C.C.C> are all considered to be identical where * represents unresponsive hop. This change to the route comparisons reduces the percentage of heterogeneous /24 blocks, but very slightly: The percentage of heterogeneous blocks decreases to 87% from 88%.

2.2 Per-destination load-balancing matters

The unexpectedly high ratio of heterogeneous /24 blocks implies that there can be other confounding factors than per-flow load-balancing addressed by Paris-traceroute MDA and ICMP rate limiting. One possibility is that load balancing is performed by destination, not by flow. We estimate how significant the effect of per-destination load-balancing can be.

Although Paris-traceroute MDA is used to discover per-destination load-balanced paths [6], it just enumerates all distinct paths towards the addresses within /24 blocks, assuming that paths towards the addresses within /24s are "identical" unless they are load-balanced paths.

[2]We used a single snapshot taken on the day before our measurement started. About 376M addresses were active in the snapshot we used. Since the availability of Internet hosts varies over time, some of the identified active addresses might not respond in our measurement, and there might be some other active addresses than the identified ones.

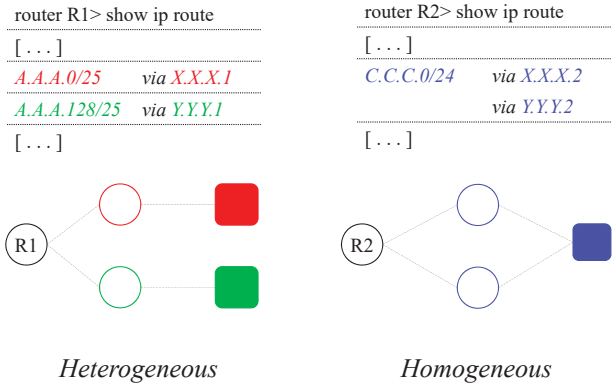


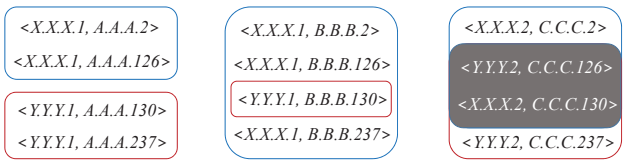
Figure 1: Different last-hop routers due to distinct route entries (left) and per-destination load-balancing (right).

However, our goal in this paper is to verify whether /24 blocks are homogeneous and thus we cannot rely on MDA. Instead, we make an assumption that is much more likely to be true than the assumption of Paris-traceroute MDA. We assume that the addresses within "/31" blocks have identical routes unless per-destination load-balancing occurs. Based on this assumption, we select two addresses that are within a /31 block from each /24, and then discover routes between a source (located at UMD) and the selected addresses using Paris-traceroute MDA. If the addresses within /31s have distinct routes, we consider that the /24s they are chosen from are affected by per-destination load-balancing. About 77% of the /31s have distinct routes. This shows that per-destination load-balancing is prevalent and can be a significant confounding factor in determining the homogeneity by comparing routes.

2.3 Dealing with per-destination load-balancing

Per-destination load-balancers can take different paths even for topologically co-located addresses. Hence, in the presence of per-destination load-balancing, homogeneity cannot be measured by simply comparing routes. A remedy is to focus on last-hop routers[3] instead of the entire routes. If routes are different due to load-balancing but eventually converge, last-hop routers will be identical. If routes are identical, last-hop routers are obviously identical. One missing case is when routes are different due to load-balancing but do not converge. In other words, last-hop routers are different due to load-balancing. It might be questionable how often this happens. According to the traceroutes dataset we collected for the addresses within /31s, about 30% of the address pairs within /31s have distinct last-hop routers. These differences are likely due to load-balancing (under the assumption that addresses within /31s are unlikely to have different routes without load-balancing).

[3]Last-hop routers are the last routers in the paths to the destinations. Their addresses may not be identified by traceroute if they do not respond to traceroute probes.



(a) Disjoint (b) Inclusive (c) Non-hierarchical

Figure 2: The relationship between the sets of the addresses grouped by last-hop routers. $< X, Y >$ denotes X is a last-hop router of a destination Y.

The question is how to distinguish whether the difference in last-hop routers is caused by load-balancing or heterogeneity. We consider that addresses are heterogeneous (in terms of topological proximity) if their last-hop routers are different due to distinct route entries rather than load-balancing[4] (figure 1). Route entries are typically generated for subnets of which network prefixes do not overlap each other, unless one subnet includes the other. Therefore, the relationships between distinct route entries will be hierarchical. To be specific, every pair of the entries will be either mutually disjoint (a sibling relationship[5]), or one includes the other (a parent-child relationship). Hence, if last-hop routers are different due to distinct route entries, when grouping addresses by their last-hop routers and representing each group by the range from the numerically smallest address in the group to the largest one, the relationships between the ranges also will be hierarchical (Figure 2a and 2b). The contrapositive of this statement, which should be also true, is that the addresses within /24 blocks are not heterogeneous (i.e., homogeneous), if any of the addresses is not hierarchical when grouped by their last-hop routers (Figure 2c). Combining this with that /24 blocks are homogeneous if their addresses have identical last-hop routers, we determine that /24s are homogeneous if any of the addresses within them does not a have hierarchical relationship with others, or they all have common last-hop routers. We call this methodology *homogeneous block identification technique* (Hobbit).

3 ELABORATION ON HOBBIT

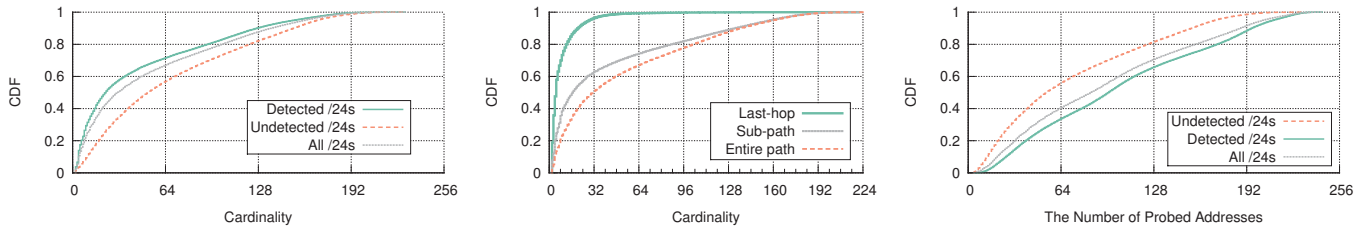
3.1 Last-hop vs entire traceroute

The basic idea of Hobbit is to examine whether the addresses within /24s have hierarchical relationships. This idea is applicable not only to last-hop routers but also to entire traceroutes. (We can group addresses having common traceroutes and check the relationships between the groups.) Nevertheless, we focus on last-hop routers. Reducing measurement loads (as we describe in Section 3.4) is not the only reason. More importantly, the coverage of Hobbit is enhanced when applied to last-hop routers compared to when applied to entire traceroutes. We compare how many homogeneous /24s Hobbit finds in each case. /24 blocks that have /31s of which traceroutes are different are likely to be homogeneous. Among these, we only select the /24s having different last-hop routers for fair comparison. If all the last-hop routers of a /24 are the same, we can conclude that it is a homogeneous block without checking the relationships, which is an advantage for the case of when applied to last-hop routers. We collect the traceroutes of all the active addresses within the chosen /24s (from a machine at UMD using Paris-traceroute MDA). We then apply Hobbit using two metrics, last-hop routers and entire traceroutes. In terms of traceroutes, only 70% of the /24s were determined to be homogeneous, which is quite low considering that we only selected /24s that are likely to be homogeneous. On the other hand, 92% of the /24s were homogeneous in terms of last-hop routers. We investigate what causes the difference.

Load-balancers use hashing to determine the next hop. Thus there is a chance that load-balanced paths appear to have hierarchical relationships. If this false hierarchy appears, Hobbit may fail to recognize the homogeneity. The question is how often hashing falsely suggests hierarchy, what it is related to, and how Hobbit can control it. We observe that its probability is closely related to cardinality, that is, the number of distinct traceroutes (or last-hop routers) towards the addresses within /24. Figure 3a shows the CDF of the cardinalities (in terms of traceroutes) of the homogeneous /24s that were detected and undetected by Hobbit (along with those of all the homogeneous /24s). We can see that the undetected homogeneous /24s tend to have higher cardinalities compared to the detected and all homogeneous /24s. This implies that cardinality influences the probability of failures. The cardinality of /24s varies a lot depending upon the metrics that define cardinality. Figure 3b shows the CDF of the cardinalities of all the homogeneous /24s in terms of traceroutes, last-hop routers and sub-paths which indicate the paths from the routers that are common to all the destinations within /24 and closest to the /24. As we use smaller parts of traceroutes, cardinality tends to decrease. One reason could be that there are multiple load-balancers on the paths. The cardinality multiplicatively increases as the number of load-balancers increases. For example, if load-balancers L_1 and L_2 distribute traffic across N_1 and N_2 paths, the total number of distinct paths can be up to $N_1 * N_2$. In comparison to the cardinalities of entire traceroutes, those of last-hop routers are very small, and this is why the coverage of Hobbit is enhanced by 22% when using last-hop routers compared to using traceroutes.

[4]Per-destination load-balancing is often implemented by installing route-cache entries for each of destinations [9]. We do not consider them to be distinct. We only consider route entries for different destination networks to be distinct.

[5]We use the term "sibling" in that distinct subnets within a /24 subnet have a common /24 prefix (i.e., a common parent).



(a) (b) (c)

Figure 3: CDF of (a) cardinality and (c) the number of probed addresses for detected and undetected homogeneous /24 blocks by our methodology. (b) CDF of cardinality in different metrics, entire traceroute, last-hop router and sub-path which indicates the path from a common router to destination.

3.2 How many destinations need to be probed?

Although Hobbit may fail to detect some homogeneous blocks depending on the cardinality, the probability of failures can be controlled by probing more destinations (because the probability is related to the number of probed addresses as shown in Figure 3c). The question is how many destinations need to be probed for a certain confidence level. We decide the number of destinations, by computing the probability of failures for each <cardinality, number of probed addresses> pair. Although the probability function could be theoretically developed, we rely on empirical analysis in this paper. We use the traceroute dataset collected for all active addresses within homogeneous /24s (as described in Section 3.1). For every combination of the destinations within a homogeneous /24, we can predict whether Hobbit will determine the /24 to be homogeneous if it only probes the destinations corresponding to the combination (simply by applying Hobbit to the partial information corresponding to the combination). All the combinations that would be determined not to be homogeneous are failures (and the others are successes), because all the combinations are chosen from homogeneous /24s. By classifying combinations by the number of destinations within them and cardinality (and computing the failure ratio in each category), we can obtain the probability of failures for each <cardinality, number of probed addresses> pair. One issue is that the total number of combinations is excessive. (The number of combinations for each /24 is $\sum_{i=1}^{n} \binom{n}{i}$ where n is the total number of active addresses within the /24, and we have data for more than 150k /24s.) To deal with this, we choose a random sample of all combinations such that most of the <cardinality, number of probed addresses> pairs have at least 16,588 sample points[6]. Figure 4 depicts[7] the distribution of degree of confidence, that is, 1 - failure ratio. As expected, the confidence tends to increase as the number of probed addresses increases and cardinality decreases. We use this data in deciding when to stop probing (as detailed in Section 3.5).

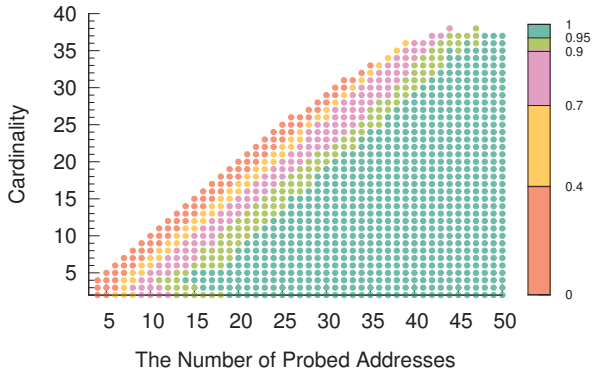


Figure 4: Degree of confidence that Hobbit will recognize a homogeneous /24 block per <cardinality, the number of probed addresses> pair.

3.3 How to select destinations?

Hobbit requires at least 4 active addresses to be effective. It is because the relationships between less than 4 addresses are always hierarchical no matter how they are grouped. We also require that every /26 within /24 has at least one active address, so that our result represents the entire /24. We identify all active addresses using ZMap ICMP Echo Request dataset, and only select /24s that meet the criterion. For each chosen /24, we group the active addresses within it by their /26 prefixes, and then probe each /26 group in a round-robin fashion. We shuffle the order of the /26s to probe at the end of each round.

3.4 How to identify last-hop routers?

The only information we need to gather by probing the destinations are their last-hop routers. In order to efficiently identify last-hop routers, we try to infer a hop count between source and a last-hop router. We send an ICMP Echo Request to a destination and inspect the response's TTL field. If we know a default TTL value of the destination host (that is, the initial TTL value written by the destination host), we can compute the hop count between source and destination. Although

[6]We obtain this number by computing the number of samples required for 99% confidence level, 1% margin of error, 50% sample proportion estimate and infinite population size [10].

[7]The values of some pairs were not depicted because they have less than 16,588 sample points at a chosen sampling rate.

default TTL values are different for different operating systems, the values of 64, 128 and 255 are commonplace [11, 12, 13]. So we consider that a default TTL is 64 if the TTL value of the response (TTL_{res}) is less than 64. If $64 \leq TTL_{res} < 128$, $128 \leq TTL_{res} < 192$ or $192 \leq TTL_{res}$, a default TTL is considered to be 128, 192 or 255, respectively. Once we identify the default TTL value of a destination, we compute the hop count between source and the last-hop router by subtracting the TTL_{res} from the default TTL value. We then run Paris-traceroute MDA with the $first\_ttl$ configured to the hop count. The inferred hop count value may be inaccurate if routers use customized default TTL values or the hop counts of the forward and reverse path are different. If the hop count is an underestimate, we will find some more routers than the last-hop router. If the hop count is an overestimate, we will fail to identify the last-hop router. If it happens, we halve the $first\_ttl$ and run again Paris-traceroute MDA. This is repeated until the last-hop router is identified or the $first\_ttl$ becomes 1.

3.5 When to terminate?

Hobbit determines that a /24 is homogeneous if all the addresses have a common last-hop router, or any of them have a non-hierarchical relationship when grouped by their last-hop routers. Hence, we terminate probing if the non-hierarchical relationship is found or we can confirm that all the addresses have a common last-hop router. To determine with a high degree of confidence whether or not a /24 has a single last-hop router, we exploit the analysis of Paris-traceroute MDA. That is, a router has a single nexthop interface (for a certain destination) at the probability of 95% if 6 probes are responded by a single nexthop interface [14]. We can view the number of interfaces as a random variable and thus substitute it with the number of last-hop routers. Therefore, we determine that a /24 has a single last-hop router (and stop probing), if we only find a single last-hop router having probed 6 destinations. We also terminate probing when we have probed as many destinations as required for 95% confidence level (figure 4). If no confidence value is present for the current <cardinality, number of probed addresses> pair, we probe all the active addresses.

4 MEASUREMENT RESULTS

We measure the homogeneity of /24 blocks using Hobbit. We choose 3.37M /24 blocks based on the ZMap data (Section 3.3), and probe each of them from a machine located at UMD[8]. In this section, we present and analyze the measurement results.

[8]We probed 64.45M destinations in total, and 54.05M were responsive. We used a single machine (at UMD) as a source, and generated a single snapshot.

Classification		# of /24 blocks
Not analyzable	Too few active	840,258 (24.9%)
	Unresponsive last-hop	567,439 (16.8%)
Homogeneous	Same last-hop router	616,719 (18.2%)
	Non-hierarchical	1,153,628 (34.2%)
Different but hierarchical		198,292 (5.9%)

Table 1: Measurement results of the homogeneity of /24

4.1 How homogeneous are /24 blocks?

Table 1 shows a summary of measurement results. There have been /24 blocks that were not analyzable by Hobbit. Although we only choose /24s having at least 4 active addresses using the ZMap data, some blocks had less than 4 active addresses when we probed them. Even when blocks have at least 4 active addresses, if the number of active addresses are less than required for achieving a desired confidence level, that is, 95% (figure 4), we classify the blocks as "Not analyzable". These two cases account for about 25% of the /24s we probed. Despite the large enough number of active addresses, 16.8% of the /24s were not analyzable because none of their last-hop routers were responsive.

We have found 1.77M homogeneous /24 blocks. About 0.62M blocks had common last-hop routers, and 1.15M blocks had different last-hop routers but their relationships were non-hierarchical. This result reinforces that per-destination load-balancing is prevalent and it even changes last-hop routers of destinations, and thus simply checking whether addresses have a common last-hop router is not enough for determining homogeneity. The remaining 0.2M blocks consist of the addresses that have different last-hop routers of which relationships are hierarchical. Since we probed as many addresses as required for 95% confidence level, the probability of these blocks being homogeneous is less than or equal to 5%. If we consider all these blocks as heterogeneous, we can conclude that 1.77M out of 1.97M /24s, that is, 90% of the /24s are homogeneous.

4.2 Analyzing heterogeneous /24s

Strictly speaking, the last category in table 1, a set of /24s that have different last-hop routers but the relationships of their addresses appear to be hierarchical is a mixture of homogeneous and heterogeneous /24 blocks. There is a non-negligible chance (5%) that the /24 blocks in the category are homogeneous. We have examined this category to discover /24s that are "very likely" to be heterogeneous, and found the criteria that define a certain class of /24s that are "very likely" to be heterogeneous.

The first criterion is that, when the addresses within /24 are grouped by their last-hop routers, the relationship between any pair of the groups is disjoint (i.e., not inclusive). Second, the groups are aligned. To be specific, when each group is represented by a subnet whose network prefix is the longest common prefix of the ad-

Composition	Ratio
{/25, /25}	50.48%
{/25, /26, /26}	20.65%
{/26, /26, /26, /26}	15.79%
{/25, /26, /27, /27}	5.92%
{/26, /26, /26, /27, /27}	4.63%
{/26, /26, /27, /27, /27, /27}	1.13%
{/25, /26, /27, /28, /28}	0.81%
{/25, /27, /27, /27, /27}	0.58%

Table 2: The distribution of homogeneous sub-blocks within heterogeneous /24 blocks

dresses within group, every subnet contains only the addresses that are within the corresponding group. For example, if we observe that the addresses <X.Y.Z.2, X.Y.Z.125> and <X.Y.Z.129, X.Y.Z.254> have common last-hop routers respectively, then we will consider that X.Y.Z.0/24 is a heterogeneous block, because the two groups are disjoint and the two corresponding subnets, X.Y.Z.0/25 and X.Y.Z.128/25 only contain the addresses within each group. If the second group were <X.Y.Z.127, X.Y.Z.254>, we would not consider this /24 to be heterogeneous because the groups would be disjoint but not aligned. /24 blocks that satisfy this criteria are very likely to be heterogeneous. We verified that homogeneous /24 blocks meet the criteria at the probability of less than 0.1%. Based on this criteria, we found 17,387 heterogeneous /24 blocks (in other words, the other 198,292 - 17,387 /24 blocks were either inclusive or disjoint but not aligned). These blocks consist of homogeneous sub-blocks. Table 2 shows the distribution of sub-block compositions. More than half of the /24s are composed of two homogeneous /25 blocks. One /25 along with two /26s and four /26s are also common compositions. /27 and /28 are also present although they are not as common as /25 and /26.

Given that at least 90% of the /24s are homogeneous, it could be considered unusual to split /24s into smaller sub-blocks and treat them differently. In order to discover who is splitting /24 blocks and why, we obtain AS numbers, organization names and geolocations of all the heterogeneous /24s using the Maxmind GeoLite database [15]. We then group the /24 blocks by the ASN they belong to. Table 3 shows the top 10 ASes with the most number of heterogeneous /24 blocks, along with organization names, countries the /24s have been allocated to, and the types of organizations we figured out from their websites. The top 2 ASes, which are both from Korea, include about 60% of the heterogeneous /24s. Other countries also tend to have more than one AS. France, Denmark and Georgia each have two. The US has one AS of which organization type is a hosting company; the rest are under the control of broadband ISPs.

To further analyze heterogeneous /24 blocks, we make

Rank	# of Heterogeneous /24s	ASN	Organization	Country	Type
1	8207	AS4766	Korea Telecom	Korea	Broadband ISP
2	1798	AS9318	SK Broadband		
3	499	AS15557	SFR	France	
10	106	AS35632	IRIS 64		
4	486	AS3292	TDC A/S	Denmark	
6	172	AS9158	Telenor A/S		
5	242	AS4788	TM Net	Malaysia	
8	115	AS28751	Caucasus	Georgia	
9	108	AS20751			
7	125	AS36352	ColoCrossing	US	Hosting

Table 3: Top 10 ASes having the most number of heterogeneous /24 blocks

IPv4 Address	: 220.83.88.0/25	220.83.88.128/26	220.83.88.192/26
Organization Name	: KT	Chungbukbonbujang	Donghajeongmil
Network Type	: CUSTOMER		
Address	: Cheongwon-Gu	Jincheon-Eup	Munbaek-Myeon
	Cheongju-Si	Jincheon-Gun	Jincheon-Gun
Province	: Chungcheongbuk-Do		
Zip Code	: 360172	365-800	365-860
Registration Date	: 20160112	20150317	20150317

Table 4: WHOIS responses from KRNIC for a /24

WHOIS queries to KRNIC [16], which is a Korean national Internet registry maintaining specific information about the addresses allocated to Korea. We focus on the top AS, Korea Telecom, because it keeps assignment information current. We made a query for each of the heterogeneous /24s and could verify that they are actually being split into sub-blocks. Table 4 shows an example. The /24 block 220.83.88.0/24 is divided into 220.83.88.0/25, 220.83.88.128/26 and 220.83.88.192/26, each of which is allocated to different customers located at different addresses. Although Korea has more than 100 million IPv4 addresses [17], considering that nearly all the heterogeneous blocks including the example block have been registered in 2015 or later, IPv4 address depletion might be a reason for splitting the /24 blocks.

5 AGGREGATING IDENTICAL /24S

A natural extension of the measurement of the homogeneity of /24s is to find homogeneous sub-blocks within heterogeneous /24s and to find larger homogeneous blocks than /24s (by aggregating them) if they are homogeneous. In this section, we focus on the aggregation of homogeneous /24 blocks.

We associate each homogeneous /24 with the set of last-hop routers of the addresses within the /24. The set can be a singleton if all the addresses within a /24 have a single common last-hop router, but can instead include multiple last-hop routers if the addresses have different last-hop routers due to load-balancing. An obvious approach to aggregation would be to merge /24s

Rank	1	2	3	4	5	6	7
Cluster Size	1251	1187	1122	1071	940	857	840
ASN	AS18779	AS1257	AS16509	AS2914	AS32392	AS1257	AS4713
Organization	EGI Hosting	Tele2	Amazon	NTT America	OPENTRANSFER	Tele2	OCN
Geo-location	US	Sweden	Japan	US	US	Sweden	Japan
Type	Hosting	Broadband ISP	Hosting/Cloud	Hosting/Cloud	Hosting	Broadband ISP	Broadband ISP

8	9	10	11	12	13	14	15
835	783	732	731	703	699	698	679
AS16509	AS4713	AS9506	AS17676	AS26496	AS22394	AS32392	AS22773
Amazon	OCN	SingTel	SoftBank	GoDaddy	Verizon Wireless	OPENTRANSFER	Cox
US (San Jose)	Japan	Singapore	Japan	US	US	US	US (Arizona)
Hosting/Cloud	Broadband ISP	Broadband ISP	Broadband ISP	Hosting	Mobile ISP	Hosting	Fixed ISP

Table 5: Top 15 largest homogeneous blocks

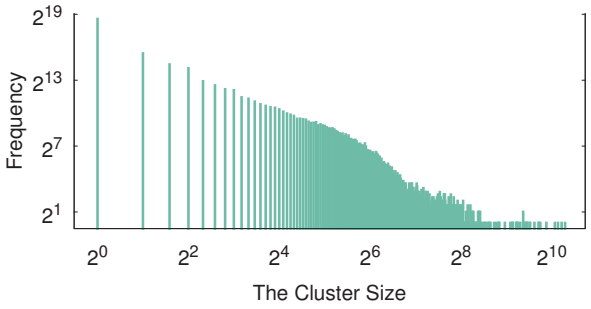


Figure 5: The size distribution of aggregated homogeneous blocks in terms of /24 blocks they contain

having the identical[9] sets of last-hop routers. In this section, we present the aggregation results made using this straightforward method. (In the next section, we consider /24 blocks that have overlapping but not identical sets, which may or may not be homogeneous.)

One may consider to apply the basic idea of Hobbit (Section 2.3) to larger blocks than /24, rather than to aggregate identical /24s. However, given the sets of last-hop routers, it is more obvious to aggregate /24s having the identical sets than checking hierarchy.

5.1 How large are the aggregated blocks?

By aggregating homogeneous blocks that have identical sets of last-hop routers, the total number of homogeneous blocks has been reduced from 1.77M to 0.53M (including not aggregated homogeneous /24s). Figure 5 depicts the distribution of size; that is, the number of /24s within the aggregated blocks. About 0.39M blocks have the size of 1, which indicates that they have not been aggregated. Still, many blocks have the size greater than 1. Although the number of blocks with the size x decreases as x increases, 21,513 blocks consist of at least 16 /24s, and 2,430 blocks contain at least 64 /24s. There are even blocks that include more than 1024 /24s. This result demonstrates that, even though /24 blocks are mostly homogeneous, they are not necessarily the largest homogeneous block. Therefore, using /24s could be inefficient. For example, since traceroutes towards homogeneous addresses are likely to be the same, selecting destinations for topology discovery from each /24 might be less efficient than choosing the destinations from the homogeneous blocks we have identified.

[9]We consider two sets are identical if their sizes are equal and every last-hop router in one set is also in the other set.

5.2 Who are the biggest homogeneous blocks?

In the presence of IPv4 address exhaustion, assigning a large number of addresses to the machines located at the topologically same location may seem unexpected. To understand why it happens, we characterize top 15 largest homogeneous blocks. We identify their ASNs, organization names and geolocations using the Maxmind GeoLite databases, and the types of organizations from their websites. Table 5 summarizes the identification results. With respect to their types, "Hosting" indicates a hosting company. We add the suffix "/Cloud" to "Hosting" if the website describes their hosting services as cloud computing services. Although Amazon is well-known for electronic commerce, we classify it as "Hosting/Cloud" because the reverse DNS names of the addresses within the corresponding blocks begin with "ec2" which is the name of its cloud computing service. "Broadband" denotes an ISP that provides both mobile and fixed broadband services. Verizon Wireless (also known as Cellco Partnership) and Cox are classified as "Mobile Broadband" and "Fixed Broadband", respectively, because they provide each of the services only.

7 of the 15 blocks are being used by hosting companies. It is understandable that hosting companies allocate many addresses to the same region because they run datacenters for their services. The addresses within each block might have been assigned to the servers in a

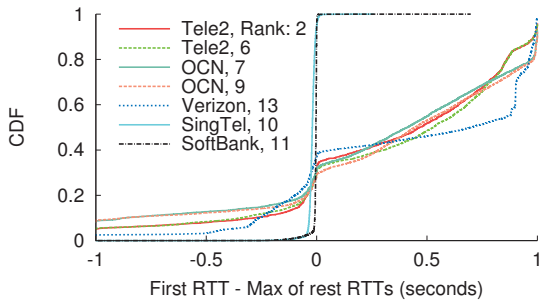


Figure 6: The CDF of the differences between the first RTT and the maximum of the rest RTTs for "broadband" blocks

datacenter. Actually, the two blocks of Amazon appear to be allocated to their datacenters. The reverse DNS names of the addresses within each block have the common keyword "ap-northeast-1" and "us-west-1", respectively, which indicate the endpoints of their datacenters located in Japan and US west [18].

6 blocks have been classified as "Broadband". Since "Broadband" ISPs provide both mobile and fixed broadband services, the addresses within these blocks could be allocated to cellular networks. A recent study on timeouts has observed that, if an initial probe to a destination experiences a higher delay than subsequent probes, then the destination is likely a cellular wireless device [19]. We use this observation to identify whether the addresses within each block are assigned to cellular devices. We randomly choose 200 /24s from each block, and then send 20 ping probes to every active address within the chosen /24s. For each address, we compute the difference between the RTT of the first ping and the maximum RTT of the rest of the pings. If the addresses within a block tend to have higher first RTTs than the maximum RTTs of the rest (i.e., if the differences tend to be positive), then the block is likely being used for a cellular network. Figure 6 depicts the distributions of the differences of the 6 "Broadband" blocks plus the Verizon wireless block which we add for reference. Tele2 and OCN each have two blocks and the differences tend to be high in all the blocks. About 50% of the addresses within the blocks have the differences greater than 0.5s and the differences of at least 10% of the addresses are greater than or equal to 1s. Verizon wireless also has a similar distribution. Therefore, the Tele2 and OCN blocks as well as the Verizon wireless block are likely being assigned to cellular networks. SingTel and SoftBank are very different from the others. Most of the differences are nearly zero, which indicates that they are not being used for cellular devices.

Recent studies have shown that major US cellular carriers connect their cellular networks with the Internet through a few infrastructure locations (so-called ingress points) [20, 21]. This means that probes for many cellular devices traverse a common ingress point, and thus they would appear to be co-located on the Internet topology. This explains why Verizon wireless has a large homogeneous block. We suspect that the ingress points of Tele2 also cover a wide area because the addresses within the Tele2 blocks are located across three countries– Sweden, Croatia and Netherlands, according to the Maxmind GeoLite databases and their reverse DNS names. We are not certain that OCN ingress points also cover a wide area but it appears likely, considering that the OCN blocks are as large as Tele2 and Verizon wireless blocks. Therefore, our result may imply that not only US cellular carriers but also European (Tele2) and Asian (OCN) carriers deploy only a few ingress points.

The last block is owned by Cox, which provides fixed broadband service to residential and business customers. Most of the addresses within the block are located in Phoenix, Arizona according to the Maxmind GeoLite databases and their reverse DNS names. They do not seem to be residential addresses, considering that most of their reverse DNS names begin with "wsip" whereas Bitcoin nodes in the Cox network (which are likely to be residential) mostly have the reverse DNS names beginning with "ip" [22]. Cox operates a large datacenter in Phoenix for business customers [23]. It could be the location where the addresses within the Cox block are allocated to. Singtel and SoftBank also provide datacenter services. The Singtel and SoftBank blocks might also be assigned to datacenters, considering that their RTTs were very stable (Figure 6).

5.3 Are the addresses within blocks numerically adjacent?

Topologically co-located addresses may be expected to be numerically adjacent because routing decisions are usually based on prefixes rather than the entire address. In this section, we analyze the numerical adjacency of the /24 blocks within the homogeneous blocks we have identified. We estimate the degree of adjacency between a /24 pair by computing the longest common prefix length of the pair. Since we compare /24s (rather than entire addresses), the length ranges from 0 to 23, and high length represents high degree of adjacency.

We numerically sort the /24s within each homogeneous block, and then compute the common prefix length between the /24s that are right next to each other. Figure 7a shows the distribution of the lengths. More than 30% of the /24 pairs have the length 23, and the lengths of about 70% are at least 20. This implies that many /24s are contiguous within the blocks. However, this does not necessarily mean that the blocks mostly consist of a single contiguous block. We next measure the common prefix length between the smallest and the largest /24s within each block (figure 7b). About 40% of the pairs have the length 0 or 1 whereas only about 5% have the length 23. This, in combination with the above result that many /24s are contiguous, implies that homogeneous blocks often consist of multiple contiguous sub-blocks that are separated from each other.

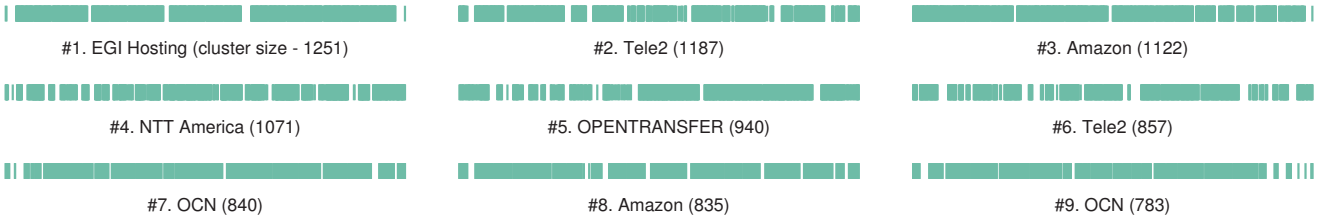
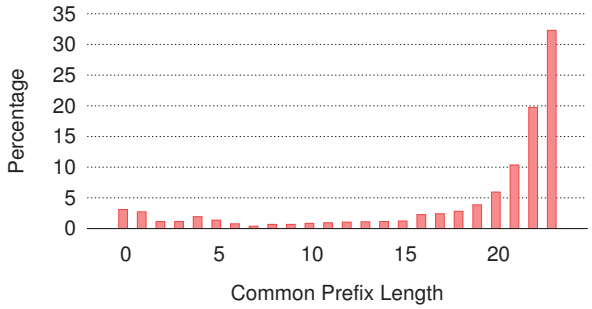
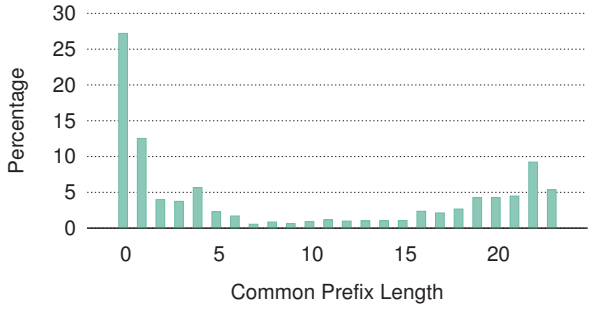


Figure 8: Visualization of numerical adjacency of /24s within the top 9 homogeneous blocks



(a)



(b)

Figure 7: The length distribution of the longest common prefixes between (a) adjacent /24s within homogeneous blocks (b) the smallest and the largest /24s

We verify that homogeneous blocks consist of contiguous sub-blocks by visualizing the adjacency of the /24s within the top 9 largest homogeneous blocks in Figure 8. For each block, given a sorted list of /24s $\{p_1, p_2, ..., p_n\}$, we draw a vertical line at x_i such that

$$x_i = \begin{cases} 1 & \text{if } i \text{ is } 1 \\ x_{i-1} + (24 - LCP\_LEN(p_{i-1}, p_i)) & \text{if } i > 1 \end{cases}$$

where $LCP\_LEN(p_i, p_j)$ denotes the longest common prefix length of p_i and p_j. The gap between the vertical lines represents the degree of adjacency. A large gap indicates low degree because the gap becomes larger as the length of the corresponding longest common prefix decreases. Most of the blocks contain large contiguous segments, none of which covers the entire block. This demonstrates that large homogeneous blocks mainly consist of several contiguous sub-blocks that are separated from each other.

6 AGGREGATING DIFFERENT BUT SIMILAR /24 BLOCKS

Aggregating the /24 blocks that have identical sets of last-hop routers is an all-or-nothing approach. /24 blocks having some common last-hop routers but not identical are treated the same as disjoint /24 blocks, irrespective of how many last-hop routers they have in common. This could be too conservative, because some /24s that are actually homogeneous may appear by measurement to not have identical sets of last-hop routers. In this section, we explain when they appear so and discuss how to aggregate those /24s.

6.1 Why care about similar blocks?

/24 blocks are associated with multiple last-hop routers if the addresses within them have different last-hop routers due to per-destination load-balancing. Unlike per-flow load-balanced paths that can be enumerated by controlling the header fields of the probes towards a single destination, per-destination load-balanced paths are found by sending probes to distinct destinations. Therefore, we may fail to identify some load-balanced paths (and thus some last-hop routers) of a /24 if it has a small number of responsive addresses. As a result, some homogeneous /24s might have non-identical sets of last-hop routers. Probing /24s varying vantage points and times can alleviate this problem, because some routers compute hashes for per-destination load-balancing based on both the source and destination IP address [24] and the availability of /24 blocks varies over time [2]. However, the measurement load of this approach can be very heavy depending on how many times we repeat probing. Furthermore, many of the probes can be wasted, particularly for addresses that are constantly unresponsive and load-balancers that do not involve source addresses in deciding next hop. For this reason, we try a different approach. We explore the possibility of inferring the homogeneity from partial information using clustering.

6.2 Selecting a clustering algorithm

Not all clustering algorithms are applicable to our problem. For example, we cannot employ k-means clustering because it runs on points in a vector space, which is not proper to model our problem. A partitional clustering algorithm proposed for identifying homogeneous IPv4 address blocks in terms of address usage [25] is also not suitable for our problem, because it assumes that addresses are only grouped into blocks that are numer-

ically adjacent, which is not true for homogeneity in terms of topological proximity (as shown in Figure 7b).

Hierarchical agglomerative clustering supports any pairwise distance so it can be a candidate. Graph clustering algorithms are also suitable in that our problem can be modelled as a graph. Graph clustering algorithms have been shown to perform better than hierarchical agglomerative clustering [26], and MCL (the Markov Cluster Algorithm) [5, 27], which is one of graph clustering algorithms, has shown a superior performance over other graph clustering algorithms [28]. Thus, we choose MCL.

6.3 Modeling and preprocessing

Given a list of /24 blocks associated with the sets of last-hop routers, we first quantify similarities between all /24 pairs. For two /24s A and B which are associated with the sets of last-hop routers S_A and S_B, the similarity score between A and B is defined as $\frac{|S_A \cap S_B|}{\max(|S_A|, |S_B|)}$. For example, if a /24 block A has a set of last-hop routers {1.1.1.1, 2.2.2.2, 3.3.3.3} and B has {3.3.3.3, 4.4.4.4}, the similarity score between A and B is $\frac{1}{3}$. We model /24 blocks and similarity scores as a weighted undirected graph. We represent each /24 as a vertex and the similarity score between each /24 pair as a weighted edge connecting the corresponding vertices, of which weight is equal to the score. (If a /24 pair has the score 0, i.e., it has disjoint sets of last-hop routers, we do not make an edge). MCL takes the graph as an input and generates the groups of vertices that are likely to be homogeneous.

One issue is that MCL has high time and space complexity (like other graph clustering algorithms and hierarchical agglomerative clustering). It requires $O(N^3)$ time and $O(N^2)$ space where N is the number of vertices. Since we have 1.77M /24 blocks and thus the input graph with 1.77M vertices, memory and time requirements can be excessively high.[10] To mitigate this, we pre-process the input graph in two steps. First, we aggregate the vertices connected by an edge of weight 1. Since the /24s corresponding to those vertices have the identical sets of last-hop routers, their similarity scores with other /24s are also identical. Hence, we can consider those vertices as one vertex. This step also can be viewed as creating a vertex for each of the aggregated homogeneous blocks that we described in Section 5.1, instead of for each /24. Second, based on the intuition that the input graph is unlikely to be (strongly) connected (because many /24s are topologically separated), we divide the input graph into multiple connected components and run MCL separately on each of the connected components. This would not degrade the clustering results because vertices that are not reachable from each other are unlikely to be clustered in MCL. Since time and space complexity of MCL is cubic and quadratic respectively, splitting an input graph reduces memory and space requirements. Another advantage is that MCL can be applied to each component in parallel. The first step reduces the number of vertices from 1.77M to 0.53M, and the second step splits the input graph into 17,563 connected components.

[10] A perturbed version of MCL can reduce time and space complexity. Unfortunately, even with that, resource requirements for our graph were very high.

6.4 Running MCL

MCL takes a parameter that determines cluster granularity. To find a good value of the parameter, we perform a parameter sweep and choose a parameter that minimizes the percentage of edges (within clusters) whose weight is less than the median of the all edge weights. We run MCL on each of the connected components using the chosen parameter. MCL has grouped 413k vertices into 58k clusters while leaving the other 120k vertices unclustered. If we regard /24 blocks that have at least one common last-hop router as being topologically co-located, the clusters generated by MCL are all likely to be homogeneous. On the other hand, if we only consider /24 blocks having the identical sets of last-hop routers as being co-located, it is uncertain whether the MCL clusters are homogeneous. We take the latter in this paper (as we did in Section 5), and thus the MCL clusters need to be verified.

6.5 Validating MCL clusters

We verify whether /24 blocks within the MCL clusters actually have the identical sets of last-hop routers using "reprobing", which is another way of dealing with similar but non-identical /24s as discussed in Section 6.1. We select 20k /24 pairs from each cluster while choosing all the pairs if a cluster has fewer than 20k pairs, and then reprobe them. We modify the original probing strategy (Section 3.5) to improve the possibility of identifying all last-hop routers for /24s. First, we do not stop probing even if non-hierarchical relationship is found. Second, we set the maximum number of probes to the number required for enumerating all interfaces with 95% confidence [14] (which is generally higher than the number required for testing the hierarchy). We probe each of the chosen /24 pairs with the modified strategy and determine that the cluster is homogeneous if all the pairs within it have the identical sets of last-hop routers. By reprobing, about 9k clusters have been determined to be homogeneous.

We may consider replacing the original probing strategy with the modified version even for measuring the homogeneity of /24 blocks (because it could potentially increase the chance of finding homogeneous blocks larger than /24). However, the modified version incurs additional measurement loads and it may have limited benefits, in that it does not help to measure the homogeneity of /24s, and that the original strategy enables us to find a substantial amount of homogeneous blocks larger than /24 (as shown in Section 5.1). Hence, we do not replace the original strategy.

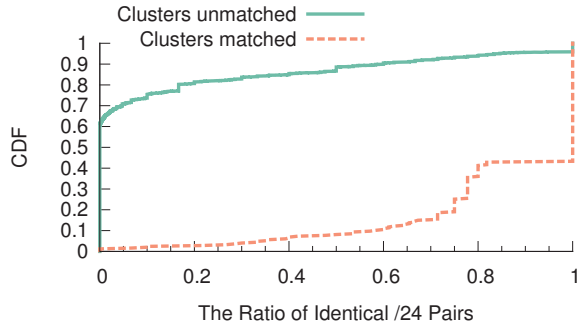


Figure 9: The ratio of identical /24 pairs within clusters that match and do not match the rule

6.6 Discussion and final results

Before presenting our final results, we discuss the possibility of inferring the homogeneity of the clusters without reprobing. In other words, we discuss if there can be a rule that distinguishes homogeneous clusters from the others. We divide the clusters in two groups, clusters confirmed to be homogeneous by reprobing and the rest. We then inspect each of the group trying to find a rule that matches only the homogeneous clusters. We have found a rule that shows promise[11]. To demonstrate the quality of the rule, we measure the ratios of the /24 pairs confirmed to have identical sets of last-hop routers by reprobing to all the reprobed /24 pairs within each cluster. A high ratio indicates a high degree of homogeneity. Figure 9 depicts the CDFs of the ratios of the clusters that match the rule and those of the others. About 90% of the clusters matching the rule have the ratio greater than 0.6 and 57% of the clusters have ratio 1. On the other hand, about 60% of the clusters that do not match the rule have ratio 0. This result shows the possibility of solely using clustering to identify homogeneous blocks. We do not include the clusters that match the rule in our final results unless they are confirmed to be homogeneous by reprobing. Our rule is experimental in that it does not match all the homogeneous clusters (found by reprobing), and the clusters matching the rule need to be further verified whether they are homogeneous.

We have found 9k additional homogeneous blocks using clustering in combination with reprobing. Figure 10 shows the changes in the distribution of the cluster sizes caused by new clusters. A substantial number of small-sized clusters (2^0-2^5) decreased, implying that they were aggregated into larger clusters. As a result, the numbers of midsize clusters ($2^5 - 2^8$) increased. The distribution of large clusters ($2^8 - 2^{11}$) also changed. A representative change is the creation of a large block whose size is 1,217 /24s. The block consists of the addresses allocated to Amazon EC2 servers located at Dublin, Ireland. To summarize, 8,931 clusters have been created

[11]The rule operates over the distribution of the similarity score between every /24 pair within a cluster. We manually built the rule.

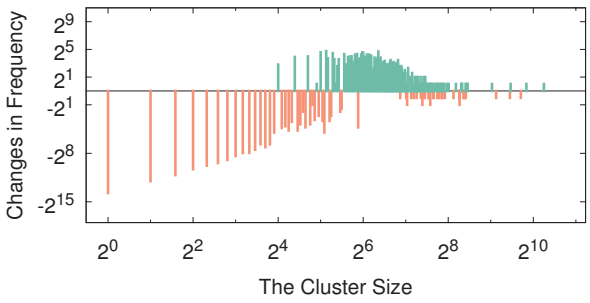


Figure 10: Changes in the size distribution of homogeneous blocks made by clustering

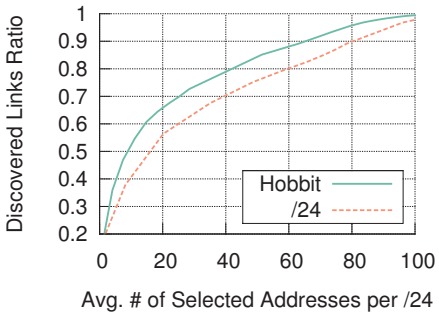


Figure 11: The ratio of the links discovered by two different approaches: To select addresses from 1) each Hobbit block and 2) each /24.

by aggregating 33,023 existing clusters, and thus the total number of clusters has been reduced from 532,850 to 508,758.

7 IMPLICATIONS

7.1 Implication for topology discovery

The homogeneous blocks identified by Hobbit can be used in selecting destinations for topology discovery. We demonstrate that measurement loads for topology discovery can be reduced if the destinations are chosen from each of the Hobbit blocks rather than each /24. We utilize the dataset collected for the comparison of the cardinality in different metrics (Section 3.1). The dataset contains the traceroutes of all active addresses within homogeneous /24s. We choose destinations (and extract the corresponding traceroutes) from the dataset in two different approaches, that is, to select a destination from 1) each /24, and 2) each of the Hobbit blocks. We then compute a discovered links ratio, that is, the number of distinct links within the chosen traceroutes divided by the total number of distinct links in the dataset. Due to the prevalence of per-destination load-balancing, selecting a single destination from each block is not enough to achieve the ratio of 1. We repeat to select more destinations from each block and recompute the ratio until the ratio nearly becomes 1. Figure 11 shows the ratios achieved by the different approaches as a function of the average number of selected destinations per /24 (that is, the number of selected destinations divided by the total number of /24 blocks

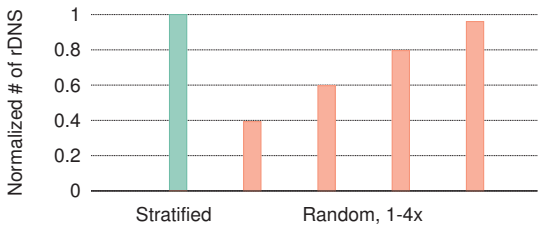


Figure 12: The normalized number of distinct rDNS patterns within samples generated by stratified and simple random sampling

in the dataset)[12]. Selecting destinations from the Hobbit blocks always results in the discovery of more links compared to selecting destinations from each /24. This demonstrates that the use of the Hobbit homogeneous blocks can improve the efficiency of topology discovery.

7.2 Implication for identifying cellular devices

Hobbit identifies some clusters consisting of cellular devices (as discussed in Section 5.2). Last-hop routers to those clusters are likely gateways that connect cellular networks to public Internet. Therefore, all the IP addresses within the clusters are very likely cellular IP addresses (i.e., IP addresses assigned to cellular devices). By generalizing the characteristics of these addresses, we can find some rules for identifying cellular devices. For example, if the addresses have a common reverse DNS (rDNS) pattern, then the pattern can be used for identifying cellular IP addresses in general.

We actually tried to find rDNS patterns for cellular IP addresses using the OCN and Tele2 blocks (that we described in Section 5.2). All the IP addresses within the Tele2 blocks had a common rDNS pattern, that is, "^m[0-9].+\.cust\.tele2" (in regular expression). The OCN blocks also had a dominant pattern. About 95% of the rDNS names had a common keyword "omed". To ensure that these patterns are only used for cellular IP addresses, we checked whether the patterns do not match the rDNS names of routers (that we identified using traceroute), and other types of end-hosts (that we identified from a list of Bitcoin nodes [22], which are not likely cellular devices). None of the rDNS names matched any of the patterns. This result demonstrates that Hobbit blocks can be used for identifying cellular devices.

7.3 Implication for sampling

The Hobbit blocks can also be used for sampling. We demonstrate that a more representative sample can be generated by selecting elements from each Hobbit block. Internet hosts are very diverse even within an ISP. We consider a sample to be more representative if it contains the elements for more types of hosts. Since it is very challenging to identify the types of the entire Internet hosts, we focus on a specific ISP, Time Warner cable. The reverse DNS (rDNS) naming schemes of Time Warner cable are open [29]. Different rDNS patterns represent different types of hosts, and thus the representativeness of a sample can be estimated by the number of distinct rDNS patterns in the sample[13]. We compare two sampling methods, a stratified sampling that draws a sample point from each Hobbit block and a simple random sampling (that is, to draw sample points from a whole population)[14]. We generate samples 25 times using each method, and compute the mean of the number of distinct rDNS patterns in each sample. Figure 12 shows the mean values which are normalized by the mean of the stratified sampling. (We do not include error bars because the standard error of the mean was negligible.) When the sample sizes are equal, a stratified sample contains 2.5 times more rDNS patterns than a random sample. Even when the sample size is doubled for random sampling, the number of rDNS patterns in the sample is only 60% of the number in a stratified sample. When 4 times as many sample points as stratified sampling are chosen for random sampling, the numbers of rDNS patterns becomes similar but the stratified sample still has slightly more of the patterns. This result shows that the stratified sampling that uses the Hobbit blocks generates a more representative sample than a simple random sampling.

We note that the stratified sample generated by drawing a single sample point from each Hobbit block only contains 73% of the entire rDNS patterns. This implies that some Hobbit blocks contain multiple rDNS patterns (in other words, rDNS patterns are not completely correlated within some Hobbit blocks). Nevertheless, stratified sampling from Hobbit blocks provides a much better trade-off between representativeness and sample sizes than random sampling (and stratified sampling from /24s).

8 RELATED WORK

There have been studies that measure the homogeneity of IPv4 address blocks. Cai et al. [25] identified homogeneous blocks using their clustering method in terms of address usage, and Quan et al. [2] have reported that most of /24 blocks are homogeneous in terms of geographical locations. Freedman et al. [30] also have quantified the geographic locality of IPv4 address blocks.

[12]We do not contain the overhead of constructing Hobbit blocks in this graph, because we intend to use Hobbit blocks not only for topology discovery but also for various other purposes, as we will discuss in this section.

[13]This approach has a limitation that the information in rDNS names may be stale.

[14]Random sampling can be nearly equivalent to stratified sampling from /24 blocks, if the addresses within /24 blocks have common rDNS patterns (because /24s are equal-sized blocks). We confirmed that, for Time Warner Cable, the number of rDNS patterns within the stratified samples from /24s are almost identical to that within the random samples.

More recently, Gharaibeh et al. [31] evaluated the geographic co-locality of /24 blocks based on delay measurements. Chen et al. [32] clustered the IP addresses of web clients that use the same local domain name server (LDNS), and measured the geographical locality of the clusters. Our work also measures the homogeneity of IPv4 address blocks, but instead we focus on topological proximity. Topologically distant hosts may however be indicative of these other metrics by having similar availability (e.g., web servers within different ASes) and similar geographical locations (e.g., residential hosts of different broadband ISPs that are in the same region).

Paris-traceroute MDA [14] enumerates all load-balanced paths between source-destination pairs. It can be also used for enumerating per-destination load-balanced paths. Hobbit also finds diverse paths towards /24 blocks. However, Hobbit concludes that the paths are load-balanced paths only when their relationships are non-hierarchical. This differentiates Hobbit from Paris-traceroute MDA.

Several studies have observed path diversity due to load balancing. Our work is relevant to these studies in that we also have observed and dealt with the path diversity. Augustin et al. [6] have observed that 39% of source destination pairs traversed per-flow or per-packet load balancers, and 70% traversed per-destination load-balancers. Flach et al. [33] have quantified violations of destination-based forwarding due to load-balancing. Pelsser et al. [34] observed a significant difference in latency between flows for the same source destination pair, which implies the existence of per-flow load-balancers.

9 CONCLUSION

In this paper, we presented the design and implementation of a methodology called Hobbit that measures the homogeneity of /24 blocks in terms of topological proximity. Hobbit deals with path diversity due to per-destination load-balancing by distinguishing between route differences due to load-balancing and different route entries. We have identified 1.77M homogeneous /24 blocks using Hobbit and aggregated them into 0.51M homogeneous aggregate blocks. We have characterized the top 15 biggest blocks, and found that most of them have been allocated to datacenters or cellular networks. We also have discovered that most of the blocks consist of numerically discontiguous addresses. Finally, we have shown that the use of the blocks for topology discovery and sampling reduces measurement loads of topology discovery and enhances the representativeness of a sample. We make the Hobbit blocks publicly available at: http://www.cs.umd.edu/~ydlee/hobbit/

As future work, we intend to apply Hobbit to IPv6 networks. We also plan to perform a longitudinal analysis of the homogeneity of /24 blocks to observe how IPv4 address exhaustion affects the address allocations. Finding other applications of Hobbit than topology discovery and sampling is also our goal.

Acknowledgments

We thank our shepherd, Ethan Katz-Bassett, and the anonymous reviewers for their helpful comments. This research was supported in part by NSF grant CNS-1526635.

10 References

[1] L. Quan, J. Heidemann, and Y. Pradkin, "Trinocular: Understanding Internet reliability through adaptive probing," in *SIGCOMM*, 2013.

[2] L. Quan, J. Heidemann, and Y. Pradkin, "When the internet sleeps: correlating diurnal networks with external factors," in *IMC*, 2014.

[3] CAIDA, "The CAIDA UCSD IPv4 Routed /24 Topology Dataset." http://www.caida.org/data/active/ipv4_routed_24_topology_dataset.xml.

[4] C. Contavalli, W. van der Gaast, D. Lawrence, and W. Kumari, "IETF Draft: Client subnet in DNS requests," Nov. 2014.

[5] S. v. Dongen, *Graph clustering by flow simulation*. University of Utrecht, 2000.

[6] B. Augustin, T. Friedman, and R. Teixeira, "Measuring load-balanced paths in the internet," in *IMC*, 2007.

[7] ZMap, "FULL IPv4 ICMP Echo Request." https://scans.io/series/0-icmp-echo_request-full_ipv4.

[8] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman, "A search engine backed by Internet-wide scanning," in *CCS*, 2015.

[9] "Cisco Load-Balancing." http://www.cisco.com/c/en/us/support/docs/ip/border-gateway-protocol-bgp/5212-46.html\#perper.

[10] S. K. Thompson, *Sampling*. John Wiley & Sons, 2012.

[11] Y.-C. Chen, Y. Liao, M. Baldi, S.-J. Lee, and L. Qiu, "OS fingerprinting and tethering detection in mobile networks," in *IMC*, 2014.

[12] B. Augustin, X. Cuvellier, B. Orgogozo, F. Viger, T. Friedman, M. Latapy, C. Magnien, and R. Teixeira, "Avoiding traceroute anomalies with paris traceroute," in *IMC*, 2006.

[13] Cisco, "TTL Expiry Attack Identification." http://www.cisco.com/c/en/us/about/security-center/ttl-expiry-attack.html.

[14] B. Augustin, T. Friedman, and R. Teixeira, "Multipath tracing with paris traceroute," in *E2EMON*, 2007.

[15] MaxMind, Inc., "Geolite databases." http://dev.maxmind.com/geoip/legacy/geolite/.

[16] KRNIC, "WHOIS services." http://whois.kisa.or.kr/eng/.

[17] KRNIC, "IPv4 address holdings." http://krnic.or.kr/jsp/eng/ipas/statistics/ipV4.jsp.

[18] Amazon, "AWS regions and endpoints."

http://docs.aws.amazon.com/general/latest/gr/rande.html.
[19] R. Padmanabhan, P. Owen, A. Schulman, and N. Spring, "Timeouts: Beware surprisingly high delay," in *IMC*, 2015.
[20] Q. Xu, J. Huang, Z. Wang, F. Qian, A. Gerber, and Z. M. Mao, "Cellular data network infrastructure characterization and implication on mobile content placement," in *SIGMETRICS*, 2011.
[21] K. Zarifis, T. Flach, S. Nori, D. Choffnes, R. Govindan, E. Katz-Bassett, Z. M. Mao, and M. Welsh, "Diagnosing path inflation of mobile client traffic," in *PAM*, 2014.
[22] "BITNODES." https://bitnodes.21.co/nodes/.
[23] Cox, "PHOENIX NAP." http://www.coxbusinessaz.com/assets/docs/BusinessContinuitySheet.pdf.
[24] Cisco, "Load Balancing with CEF." http://www.cisco.com/en/US/products/hw/modules/ps2033/prod_technical_reference09186a00800afeb7.html.
[25] X. Cai and J. Heidemann, "Understanding block-level address usage in the visible internet," in *SIGCOMM*, 2010.
[26] D. Dueck, *Affinity propagation: clustering data by passing messages*. University of Toronto, 2009.
[27] "MCL." http://micans.org/mcl/.
[28] S. Brohee and J. Van Helden, "Evaluation of clustering algorithms for protein-protein interaction networks," *BMC bioinformatics*, vol. 7, no. 1, p. 1, 2006.
[29] Time Warner Cable, "RDNS naming schemes." http://postmaster.rr.com/reverse_dns.
[30] M. J. Freedman, M. Vutukuru, N. Feamster, and H. Balakrishnan, "Geographic locality of ip prefixes," in *IMC*, 2005.
[31] M. Gharaibeh, H. Zhang, C. Papadopoulos, and J. Heidemann, "Assessing co-locality of ip blocks," in *IEEE Global Internet Symposium*, 2016.
[32] F. Chen, R. K. Sitaraman, and M. Torres, "End-user mapping: Next generation request routing for content delivery," in *SIGCOMM*, 2015.
[33] T. Flach, E. Katz-Bassett, and R. Govindan, "Quantifying violations of destination-based forwarding on the internet," in *IMC*, 2012.
[34] C. Pelsser, L. Cittadini, S. Vissicchio, and R. Bush, "From paris to tokyo: On the suitability of ping to measure latency," in *IMC*, 2013.