Understanding Block-level Address Usage in the Visible Internet (Extended)

USC/ISI Technical Report ISI-TR-665, May 2010

Xue Cai John Heidemann
USC/Information Sciences Institute {xuecai,johnh}@isi.edu

ABSTRACT

Although the Internet is widely used today, we have little information about the edge of the network. Decentralized management, firewalls, and sensitivity to probing prevent easy answers and make measurement difficult. Building on frequent ICMP probing of 1% of the Internet address space, we develop clustering and analysis methods to estimate how Internet addresses are used. We show that adjacent addresses often have similar characteristics and are used for similar purposes (61% of addresses we probe are consistent blocks of 64 neighbors or more). We then apply this block-level clustering to provide data to explore several open questions in how networks are managed. First, we provide information about how effectively network address blocks appear to be used, finding that a significant number of blocks are only lightly used (most addresses in about one-fifth of /24 blocks are in use less than 10% of the time), an important issue as the IPv4 address space nears full allocation. Second, we provide new measurements about dynamically managed address space, showing nearly 40% of /24 blocks appear to be dynamically allocated, and dynamic addressing is most widely used in countries more recent to the Internet (more than 80% in China, while less than 30% in the U.S.). Third, we distinguish blocks with low-bitrate last-hops and show that such blocks are often underutilized.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Network topology*; C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network management*

General Terms: Measurement

Keywords: Internet address usage, survey, pattern analysis, clustering, classification, availability, volatility, median-up, low-bitrate, RTT

1. INTRODUCTION

Previous Internet topology studies focused on AS- and router-level topologies [6, 8, 10, 14, 27, 31, 32]. While this work explored the core of the network, it provides little insight into the edge of the Internet and the use of the IPv4 address space. The transition to classless routing (CIDR, [13]) in the mid-1990s has made the edge opaque. Only recently have researchers begun to study edge-host behavior using server logs [40], web search engines on textual addresses [34], and ICMP probing [15].

Yet the network edge has seen great change and deserves study. How is CIDR applied? How is dynamic addressing used? How widespread are low-bitrate edge links? In this paper we use active probing to study these properties of the edge of the Internet.

Assumptions: In this paper we begin to explore the potential of *clustering of active probes to infer network address usage*. Our work makes three assumptions:

1. Many active addresses will respond to probes,
2. Contiguous addresses are often used similarly, and
3. Patterns of probe responses and response delay suggest address usage.

While there are cases where these assumptions do not hold, we believe the assumptions apply to a large fraction of the Internet and so active probing can provide insight into address usage.

We examined the first assumption and previously showed that active probes detect the majority of addresses in use, as verified with tests against a university and a random sample of the general Internet [15].

While this prior work established the collection methodology and error bounds; this paper provides the first evidence for the next two assumptions and their application to understand network usage. The second assumption is contiguous use, which follows from the traditional administrative practice of assigning blocks of consecutive addresses to minimize routing table sizes. While there is no requirement that adjacent addresses be used for the same purpose, we will show that they are often used similarly (Section 4.1).

Finally, we assume that repeated active probing with ICMP provides information about how addresses are used. We take advantage of both the pattern of positive, negative, or missing response, and the round-trip time (RTT) of the response. While a single ICMP response provides only limited information (consent of the address to reply), repeated probing can tell much more. For example, we use response patterns to distinguish intermittent from continuously used addresses, and we show that RTT can identify low-bitrate edge links.

Figure 1 shows an example of what can be learned from probing one block of 256 addresses with prefix[1] p. In this figure, the 256 addresses in prefix p are mapped into two dimensions following a Hilbert curve (each quadrant of the square shows one-quarter of addresses, recursively). Different shades indicate different ping response patterns from each address (white is non-responsive; green, availability; red, volatility; metrics we define later in Section 3.2). Two green areas are blocks of addresses that are almost always up: the single address p.65/32 at the top center, and the 32-addresses block p.128/27. The two dark areas (the lower left quarter, p.192/26, and bottom right eight, p.160/27) are used only infrequently, with low availability and volatility. We can often confirm these probe-based observations against other sources (Section 5 discusses hostnames and operator-provided ground truth). The bottom of the figure shows how we automatically identify these regions (Section 3.3).

Approach and Validation: From these assumptions we develop new algorithms to identify blocks of addresses with consistent usage (Section 3). We start with Internet survey data, where each address in around 24,000 /24 address blocks is pinged every 11 minutes for around one week [15]. From this dataset we derive several metrics about address usage. We then use these statistics to automatically identify blocks of consistent responsiveness.

Before applying these algorithms, we evaluate how often our assumptions hold. Our first question is therefore *are adjacent addresses used consistently* and *can we discover them reasonably accurately?* Before classless IP addressing [13] allocation strategies were aligned with externally visible address allocation, but since then there has been no way to easily evaluate how addresses are used. We explore these basic questions in Sections 4.1 and 5.1.1.

Applications: A first application of this approach is to understand how addresses are managed, beginning with what block sizes are typical (Section 4.1). We find that 2,529,216 addresses, or 61% of the probed address space, show consistent responses in blocks of 64 to 256 adjacent addresses (/26 to /24 blocks). Also, we observe that most addresses (around 55%) are in /24 or bigger blocks.

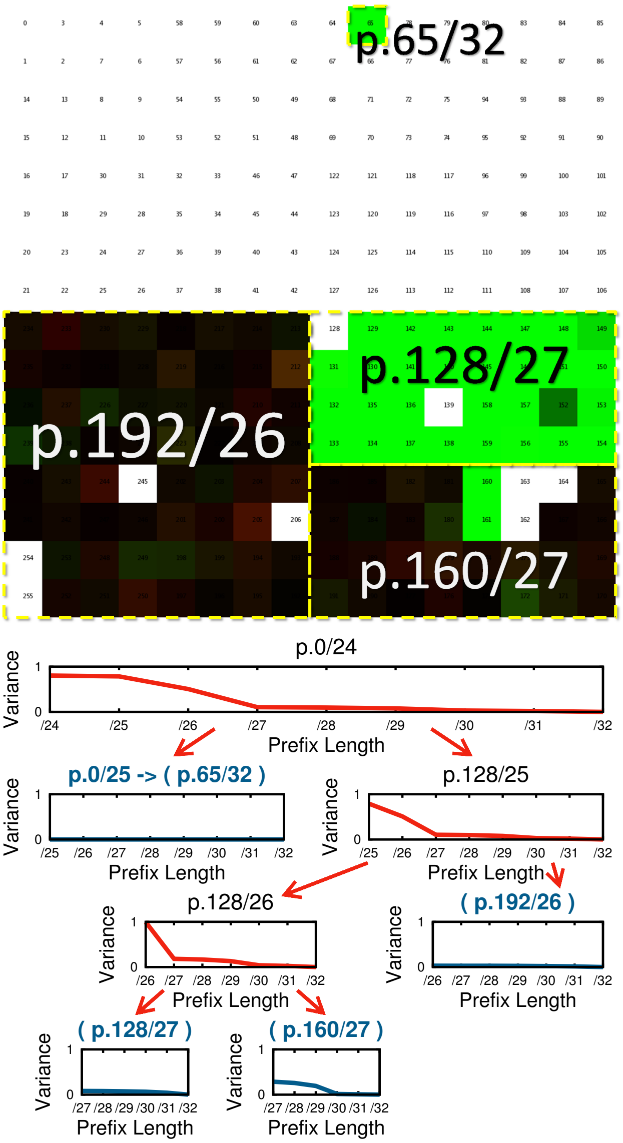

Figure 1: Top: a /24 block (prefix is anonymized to p) with 4 plausible regions of different use. Bottom: our *BlockSizeId* algorithm ($\epsilon = 2.0$) identifies these regions (Section 3.3), with best-fit variance in (parentheses).

Another application is understanding how effectively addresses are used (Section 4.2). We find that many blocks are only lightly used (about one-fifth of /24s show less than 10% utilization). Improving utilization is increasingly important as the IPv4 address space nears full allocation; improving IPv4 efficiency is a cost to compare compared to IPv6 transition.

Third, we detect and quantify the use of dynamic address assignment (Section 4.3). Dynamic addresses are

[1] Recall that IPv4 addresses are 32-bit numbers, usually written in the form $a.b.c.d$, where each component is an 8-bit portion of the whole address. Addresses are organized in *blocks* (sometimes called subnetworks) that are sized to powers of two. Blocks have a common *prefix*, the leading p bits of the address, written $a.b.c.d/p$. For example, 128.125.7.0/24 indicates a /24 block with 256 addresses in it of the form 128.125.7.x. We sometimes talk about blocks as p.0/24, where p represents the anonymized prefix.

used in some spam detection algorithms [40], and identifying dynamic addresses is important to estimate the number of computers that connect to the Internet [15]. We observe that nearly 40% of /24 blocks appear to be dynamically allocated, and dynamic addressing is much higher in countries most recent to the Internet (more than 80% in China, while less then 30% in the U.S.).

Finally, we distinguish blocks connected mainly by low-bitrate edge links from those with broadband connections, identifying blocks used by dial-up and older mobile phones (Section 4.4). Study of edge bitrate can help understand trends in technology deployment, and automatic identification of users of low-bitrate networks may allow websites to *automatically* match content and layout. Edge links and policies also interact with address utilization (Section 4.4); we show low-bitrates links are correlated with short connect-times and sparse usage.

The contribution of this paper is therefore to develop new approaches to classify Internet address usage and to apply those approaches to answer important questions in network management. As with other studies of the live Internet, our approach must employ incomplete information: our surveys cover randomly selected /24 blocks (not larger) and do not inform us about addresses that refuse to respond. However, we suggest that the approach is promising and our preliminary results provide new techniques, adding to what is currently known.

The main contents of this paper appeared as a conference paper [4]. This technical report [5] adds several appendices to provide additional data about stability of the results across different surveys and in different years (Appendix A), general characterization of the (A, V, U^*) space (Appendix B), and details about our training and hostname-inferred usage (Appendix C),

2. RELATED WORK

Much prior work has explored the Internet topology [6, 8, 10, 14, 27, 31, 32]. Recent work has begun exploring edge host behavior [15, 34, 40]. Our work builds upon this prior work and the specific work listed below.

Are contiguous addresses consistent and what are the typical block sizes? Although addresses are usually assigned as blocks and represented in prefixes by classful [28] and classless addressing [13], there is no guarantee that contiguous addresses in the same block will be used in the same way. Huston's report has analyzed the common prefix lengths in BGP routing table [16]. But it cannot look at usage at granularities smaller than BGP prefixes. Our approach is able to look at these smaller block sizes through active probing.

Are allocated addresses efficiently utilized? Several researchers have studied rates of IPv4 address consumption, predicting IANA will exhaust its allocation pool in 2011 [16]. However, full allocation does not necessarily imply full use. Prior researchers inferred address utilization by detecting allocated but not advertised prefixes in BGP routing table [25]. But what is routed may differ from what is actively used. Our work tries to track active use; and our study of individual addresses can reveal changes that happen to blocks inside an organization (smaller than are typically routed).

How many addresses are dynamically assigned? Xie et al. have begun to explore this question with a goal of identifying dynamic blocks to assist spam prevention [40]. Their work is based on passive collection of Hotmail web server logs, while our method uses a completely different approach by active probing and so can extend and corroborate their findings. In our prior work, we provide another perspective based on active probing with ICMP [15]. While this prior work focused on censuses (occasional but complete probing) and establishing the methodology, here we study survey data (frequent probing of a sample of the Internet) and add significant new analysis to identify block sizes and low-rate edges.

CAIDA has long studied Internet topology with active probing [20]. They traceroute to one address for each routed /24 address block. Our datasets differ by probing only a fraction of /24s (but all addresses in them, and much more frequently). Probing /24s allows us to take the advantage of locality to study address usage. Because contiguous addresses are usually administrated together and used in the same way, analyzing the whole block instead of sampling one address from each block can provide information not previously available. In addition, our frequent sampling shows temporal changes useful for identifying dynamic address allocation.

Regional Internet Registries (RIR) have another potential source of data, as they require organizations to state the usage of current addresses and the planned usage of new addresses [2, 3]. Such data is not generally available, but it is another possible means of future validation.

Identifying edge-link bitrates? A great deal of work has explored identification of edge-link bitrates (or link capacity) and available bandwidth. While we cannot review it all here, key results include packet pair [21] and pathchar [18]. We explore the use of variance as a new approach to estimate edge-link bitrates (Section 3.5).

3. METHODOLOGY

This section introduces our methodology: collecting raw data through an Internet survey, transforming that data into relevant observations, identifying blocks of consistent use, classifying blocks into ping-observable categories, distinguishing between low-bitrate and broadband blocks.

3.1 Data Collection: Surveying the Internet

We would like as much data about Internet addresses or hosts as possible, but we must balance that desire against today's security-conscious Internet culture. Our data collection builds on prior *Internet ICMP surveys* that ping each address of about 1% of the allocated Internet address space approximately every 11 minutes for one week or longer [15].

We use a previous selection methodology [15], selecting around 24,000 /24 blocks from those that were responsive in a prior census of all allocated addresses. We select blocks of addresses rather than individual addresses so we can study how addresses are allocated and used. Our choice of /24 blocks limits our ability to observe very large allocations, but allows the identification of blocks smaller than 256 addresses (Section 4.1). As with prior work, half of the selected blocks are kept consistent across multiple surveys and half are chosen randomly, enabling longitudinal studies while providing a subset that is selected with very little potential bias. We compare two surveys in Section 5.3, showing that our study of 1% of the address space represents a large enough fraction of the space to be representative.

Approximately every 11 minutes, each address is probed. Probes are dispersed over this period and sent in pseudorandom order to avoid correlations due to outages. Probes taken every 11 minutes limit our ability to detect very rapid churn of dynamic addresses, however prior studies of dynamic addresses placed typical use durations at 75 or 81 minutes [15,22], suggesting we have reasonable precision. Responses can be classified into three broad categories: positive (*echo reply*), negative (for example, *destination unreachable*), and non-response. In this paper we ignore all non-positive responses. Packet loss can cause measurement inaccuracy, so we use 1-loss repair to cope with singleton packet losses [15] (1-repair assumes an absent response between two consistent responses is loss and interpolates accordingly). Network outages can also distort our survey. We manually examine our survey and select a period that has no local network outages.

All surveys but *IT16ws* [35] cover more than one week, allowing us to detect diurnal and weekly cycles.

Of course, using ICMP for probing has significant limitations. The most serious is that large parts of the Internet are firewalled and choose not to respond to our probes. Some form of this bias is inherent in any study using active probing. Prior studies of a large university and a random sample of Internet addresses suggest ICMP probing undercounts hosts by a factor of 30–50%, and that ICMP is superior to TCP-based probing [15]. We recognize this limitation as fundamental to our methodology, but we know of no evidence or inference to suggest that the firewalled portions of the Internet use significantly different allocation strategies than the more open parts of the Internet. In addition, we confirm the accuracy of our results at USC (Section 5.1), and we show similar accuracy for manual inspections of blocks drawn at random from the Internet in Section 5.2. However, we are exploring additional ways to verify this assumption, and investigation of the firewalled Internet is future work.

	Start Date	**/24 Blocks**		
Name	**(# days)**	**probed**	**respond.**	**Use**
IT17ws [36]	2007-06-01 (10)	22,367	20,849	all
IT17wrs	2007-06-01 (10)	17,366	16,295	§C
IT17wvs	2007-06-01 (10)	100	100	§5.2
IT17wbs	2007-06-01 (10)	200	200	§5.2
IT16ws [35]	2007-02-16 (6)	22,365	20,900	§5.3
IT30ws [38]	2009-12-23 (14)	22,381	20,227	§5.3
IT31ws [39]	2010-02-08 (14)	22,376	19,909	§5.3
LTUSCs [37]	2007-08-13 (9)	768	299	§5.1
ISC-DS [17]	2007-01	hostnames		§5
RIR [29]	2007-06-13	block allocation		§4

Table 1: Datasets used in this paper.

Table 1 shows the datasets we use in our paper. We use two ICMP surveys taken by USC [15]: *IT17ws*[2] and *IT16ws*; *IT17ws* is the main dataset used in this paper, while we use *IT16ws*, *IT30ws*, *IT31ws* for validation in Section 5.3. Not all /24 blocks we picked respond to our pings, however, most of them did respond at least once by one IP address. We collected *LTUSCs* to compare our inferences with network operators as discussed in Section 5.1. Finally, we use a domain name survey from ISC [17] to validate our conclusions (Section 5).

3.2 Representation: Observations of Interest

Since one survey provides more than 5 billion observations, it is essential to map that raw data into more meaningful metrics. We call this step *data representation.* We define three metrics to characterize address usage: *availability*, the fraction of time an address is responsive; *volatility*, a normalized representation of how many consecutive periods the address is responsive; and *median-up*, the median duration of all up periods. And we characterize edge bitrate with two metrics: *median-RTT* and *stddev-RTT*, the median and standard deviation of RTT values of all positive responses.

3.2.1 Metrics characterizing addresses usage

To define *availability*, *volatility* and *median-up*, let $r^*_i(a)$ be the positive (1) or non-positive (0) measurements for address a (for all $i \in [1 \mathinner{..} N_p]$, where N_p is the number of probes). We analyze these values after 1-loss repair [15]:

$$r_i(a) = \begin{cases} 1, & r^*_i(a) = 1 \vee (r^*_{i-1}(a) = 1 \wedge r^*_{i+1}(a) = 1) \\ 0, & \text{otherwise} \end{cases}$$

[2] The name *IT17ws* indicates: Internet Topology, the 17th full collection, "w" collected at ISI-west in Marina del Rey, and "s" indicates a survey rather than a full census.

If each probe is made at time t_i, we can define the series of up durations of an address in a survey as

$$u_j(a) = t_{e_j} - t_{b_j},\ \forall j \in [1 \,..\, N_u] \ \text{ where}$$
$$r_i = 1, \forall i \in [b_j \,..\, e_j] \text{and } r_{(b_j)-1} = 0, r_{(e_j)+1} = 0$$

(each up duration is a consecutive run of positive probes from b_j to e_j, inclusive). There are N_u up durations in total, where $N_u < N_p$. We can now clarify that availability, volatility, and median-up are given as:

$$\begin{aligned} A(a) &= \frac{1}{N_p} \sum_{1}^{N_p} r_i \\ V(a) &= N_u / \lceil N_p/2 \rceil \\ U^*(a) &= \text{median}(u_j,\ \forall j \in [1 \,..\, N_u]) \end{aligned}$$

Availability is normalized, the fraction of times a host is reachable. Volatility is normalized by $\lceil N_p/2 \rceil$, the maximum number of states (alternating value each time). For example, if $N_p = 16$, and the responses r_i^* of address a are $[1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1]$, then firstly, we will apply 1-loss repair on r_7, because r_6 and r_8 are both positive responses. After 1-loss repair, the responses r_i are $[1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1]$. Now there are three up periods ($N - u = 3$) of lengths 22, 44, 55 minutes each. $A(a) = 11/16 = 0.688$, $V(a) = 3/(16/2) = 0.375$ and $U^*(a) = \text{median}(22, 44, 55) = 44$ minutes. (We also sometimes use un-normalized volatility, $V^*(a) = N_u$, simply the count of up periods.) We considered normalizing median-up to measurement duration, but chose not to because such normalization distorts observations about hosts that are not nearly always present.

While these metrics are not orthogonal, each has a purpose. Availability shows how effectively addresses are used. High volatility indicates addresses that are intermittently used and often dynamically allocated. Median uptime suggests how long an address is used.

These estimates assume the r_i observations are correct and represent a single host. Because we know our data collection omits firewalled hosts (Section 3.1), we generally ignore addresses that never respond. More troubling are addresses used by multiple computers at different times—such addresses actually represent *multiple* hosts. The purpose of dynamically allocated addresses is exactly to share one address with multiple computers, and we know dynamic assignment is common (see Section 4). If those hosts are used for different purposes (servers sometimes, and clients others), usage inference will be difficult and unreliable. However, we believe that it is relatively uncommon for a dynamic address to transition between client and server use, since servers usually require stable addresses. (There is some use of dynamic DNS to place services on changing addresses. We believe such use is rare for most of the world but plan to explore this issue in future work.)

3.2.2 Metrics characterizing edge bitrate

While address usage considers all ICMP responses (positive and negative), round-trip time estimates are only present in positive responses. To estimate bitrate, we therefore define $R^*(a)$ be the set of RTT values extracted from positive responses for address a, that is, the set of all $R_i^*(a)$ where $r_i^*(a) = 1, \forall i \in [1 \,..\, N_p]$. (So $|R^*(a)| \le |r^*(a)|$.) From this set we compute standard deviation of $R^*(a)$: $R^*_{\mu_{1/2}}(a)$, when we have sufficient samples ($|R^*(a)| \ge 10$).

We use these metrics to identify low-bitrate edge links. Median-RTT tracks typical response bitrate, while stddev-RTT estimates variance. In Section 3.5 these metrics can identify low-bitrate blocks.

3.3 Block Identification

We next use our observations about addresses to evaluate block size using a clustering algorithm that considers the address hierarchy.

We assume blocks are allocated in sizes that are powers of two, so block identification is the process of finding a prefix where addresses in the block are used consistently. We find that some blocks are *not* used consistently, and different addresses show very different stability. In our analysis we will keep dividing these *mixed-use blocks* until they are consistent, if necessary devolving to a single address per block. Another challenge is that many blocks have gaps where a few addresses are used differently, or are not responsive, perhaps because they are unused or firewalled. Our algorithm weighs choice of larger blocks with some inconsistencies against smaller but more homogeneous blocks.

We only consider /24 blocks and smaller because current data collection method gathers samples of that size. Exploration of larger blocks is an area of potential future work.

3.3.1 Clustering background

In clustering of address responsiveness, we want to determine blocks that appear to be used consistently.

We therefore use *partitional clustering*, one of the two general approaches to clustering in this well developed field [19]. Partitional clustering places each element into exactly one cluster; we choose it over the alternative, hierarchical clustering, which would place items into multiple, hierarchically nested clusters. Jain defines partitional clustering as: “Given n patterns in a d-dimensional metric space, determine a partition of the patterns into K groups, or clusters, such that the patterns in a cluster are more similar to each other than to patterns in different clusters” [19]. We build on the basic approaches of clustering for our method: *pattern matrix*, *feature normalization*, and using an *elbow criterion* to select the best choice.

Although we follow traditional clustering theory, Internet addresses impose a unique restriction. Addresses are only grouped into blocks that are contiguous, sizes of powers of two, and aligned at multiples of the size. For these reasons, we cannot directly use traditional algorithms such as K-means, but instead use components of existing clustering approaches. The most radical difference from traditional clustering is that addresses are only clustered with some number of immediate neighbors, not with arbitrary other addresses. We therefore find blocks of consecutive addresses by the definition of our algorithm, however the *size* of blocks it finds depends on the consistency of how the addresses are used.

A *Pattern Matrix* defines the features over the space being clustered. In our case, each address is defined by its three features $(A(a), V(a), U^*(a))$, and the space is a number of disjoint /24 blocks. (We also use $(R^*_{\mu_{1/2}}(a), R^*_{\sigma}(a))$ later in Section 3.5 to identify block connection types.) Each /24 block has a 256×3 pattern matrix x^*_{ij}, where j enumerates the three features, and i enumerates each address in a /24 block. From our 24,000 /24 blocks we get 24,000 pattern matrices in total.

Although our definitions of A and V are already normalized to the range $[0, 1]$, their distribution may be skewed, and U^* is not normalized. We therefore employ *feature normalization* to give each features equal weight. We define the normalized feature vector x_{ij}, given the mean and standard deviations m_j and s_j of each feature j:

$$x_{ij} = \frac{x^*_{ij} - \mu_j}{\sigma_j}$$

where μ_j and σ_j are the mean and standard deviation. We use Euclidean distance between two components of the feature vector to measure dissimilarity between two elements i and k over their features:

$$d(i,k) = \sqrt{\sum_{j=1}^{3} (x_{ij} - x_{kj})^2}$$

Many clustering algorithms, like K-means, require the number of clusters be chosen in advance. We cannot do that because clusters correspond to block size, a quantity we wish to discover. We also cannot simply minimize variance, because variance is trivially minimized in the degenerate case where each cluster is a singleton address.

We therefore employ an *elbow criterion*, a common rule of thumb to determine the number of clusters. We split each cluster into two whenever splitting adds significant information, and we stop when we pass the "elbow" of the curve and more clusters add little benefit. We measure information by the sum of variance in each cluster across the population—homogeneous clusters will have low variance; splitting them adds no new information. Heterogeneous clusters have high variance, and splitting them into two more self-consistent pieces reduces the sum of variance, increasing the amount of information.

We use partitional clustering [19] to determine blocks that appear to be used consistently based on their responsiveness. A *pattern matrix* defines the features of patterns (i.e., addresses) being clustered: ($A(a)$, $V(a)$, $U^*(a)$) across the space of disjoint /24 blocks. (We also use ($R^*_{\mu_{1/2}}(a)$, $R^*_{\sigma}(a)$) later in Section 3.5 to identify block connection types.) Each /24 block has a 256×3 pattern matrix x^*_{ij}, where j enumerates the three features, and i enumerates each address in a /24 block. From our 24,000 /24 blocks we get 24,000 pattern matrices in total. To give each features equal weight, we employ *feature normalization.* And we define the normalized pattern matrix as $x_{ij} = (x^*_{ij} - \mu_j)/\sigma_j$, where μ_j and σ_j are the feature's mean and standard deviation. We then use Euclidean distance to measure dissimilarity between two patterns. Because Internet addresses impose a unique restriction that addresses are only grouped into blocks that are contiguous, sizes of powers of two, and aligned at multiples of the size, we cannot directly use traditional algorithms such as K-means. We therefore employ an *elbow criterion*, a common rule of thumb to determine the number of clusters. We split each cluster into two whenever splitting adds significant information, and we stop when we pass the "elbow" of the curve and more clusters add little benefit.

3.3.2 Our algorithm to identify block sizes

Our algorithm follows the basic structure from above: we define a pattern matrix of addresses by features, normalize the features, then recursively search for clusters until reaching the elbow. We fill in the details next.

The algorithm is a recursive function, *BlockSizeId*, taking an address-feature matrix $256 \times (A(a), V(a), U^*(a))$ and a given prefix length P. Since the blocks in our survey are disjoint, we iterate over each /24 block in our survey separately, beginning with $P = 24$.

BlockSizeId then computes the intra-block unnormalized variance, $vsum_p$, for all possible prefix lengths p ($P \leq p \leq 32$). It then selects the smallest prefix length p_{elbow} where longer prefixes show minimal change.

$$\begin{aligned} n_p &= 2^{p-P}, s_p = 2^{32-p}, \mu_{bj} = \frac{\sum_{i=(b-1)s_p+1}^{bs_p} x_{ij}}{s_p} \\ v_b &= \sum_{j=1}^{3} \sum_{i=(b-1)s_p+1}^{bs_p} (x_{ij} - \mu_{bj})^2, 1 \leq b \leq n_p \\ vsum_p &= \sum_{b=1}^{n_p} v_b, P \leq p \leq 32 \end{aligned}$$

Here n_p is the number of sub-blocks with prefix length p, s_p is the size of sub-blocks (number of addresses) with prefix length p. For example, if $P = 24$ and $p = 27$, then

$n_p = 8$ and $s_p = 32$. m_{bj} is the mean value of the j^{th} feature of addresses in the b^{th} sub-block. v_b is the intra-block unnormalized variance of the b^{th} sub-block. In this example, it would be the intra-block unnormalized variance of the b^{th} /27 sub-block.

We define minimal change in the elbow algorithm with an empirically selected constant threshold, $\epsilon = 2.0$. We select p_{elbow} as the smallest p such that $vsum_{i+1} - vsum_i < \epsilon, p \leq i \leq 31$. If $p_{elbow} = P$, then *no* division of this block reduces variance significantly and we terminate our recursive algorithm, declaring P the consistent block size. If this case does not hold, we have determined there are splits of the block that appear to be more consistent. We then split the block in half and recurse, calling *BlockSizeId* with the next longer prefix $P = p+1$ on each half of the data. In principle, a block could be split repeatedly until it is composed on a single address (since singletons will drive variance to zero). In Section 4.1 we show that, in practice, our threshold causes the majority of the Internet addresses fall into larger blocks of consistent use.

3.3.3 A block identification example

To illustrate *BlockSizeId* we next show analysis of an example /24 block taken from the Internet. The top of Figure 1 shows the whole block, while the bottom graphs show how the algorithm identifies four sub-blocks. As described earlier (Section 1), a human identifies two bright green areas (or light grey) indicating high availability: p.65/32 and p.128/27, and two dark areas showing low availability and volatility, p.160/27 and p.192/26. Hostnames for this block show it is used for wireless access, and the green areas are servers and routers, while the dark areas are dynamically assigned by DHCP.

The first graph in the middle of the figure shows the first pass of *BlockSizeId*, with $P = 24$ covering all of block p.0/24. In the graph, the y-axis shows variance for division of the block into each possible power-of-two smaller size. Here $p_{elbow} = 25$ and $p_{elbow} > P$, so we recurse to $P = 25$.

The second row of two graphs shows these recursive invocations, p.0/25 on the left and p.128/25 on the right. For p.0/25 with only one responsive address, the left graph shows a consistent variance regardless of subdivision, and $p_{elbow} = P = 25$, so this prefix is consistent and this recursion terminates. For p.128/25 on the right, a subdivision reduces variance and so we recurse again to $P = 26$.

The algorithm continues until either $p_{elbow} = P$ or $P = 32$. In this example, the initial /24 block is divided into p.65/32, p.128/27, p.160/27, and p.192/26.

3.4 Ping-Observable Block Classification

We can now take remote measurements, convert them into observations, and use them to identify blocks of consistent neighboring addresses. We generalize our observations on addresses into observations about a block b by taking the median value of each observation:

$$(A(b), V(b), U^*(b)) = \text{median}(A(a), v(a), U^*(a))\ \forall a \in b$$

We then classify these blocks into five *ping-observable categories*, using $(A(b), V(b), U^*(b))$. We use four thresholds, $\alpha_H = 0.95$, indicating high availability, $\alpha_L = 0.10$, indicating low availability, $\beta = 0.0016$, for low volatility ($V(b) = \beta$ is equal to $V^*(b) = 1$, i.e., only up for once), and $\gamma = 6$ hours, corresponding to a relatively long uptime.

Always-stable: highly available and stable.

$$(A(b) \geq \alpha_H) \wedge (V(b) \leq \beta)$$

Sometimes-stable: changing more often than always-stable, but frequently up continuously for long periods (high $U^*(b)$).

$$(U^*(b) \geq \gamma) \wedge (A(b) \geq \alpha_L) \wedge (A(b) < \alpha_H \vee V(b) > \beta)$$

Intermittent: individual addresses are up for short periods (low $U^*(b)$):

$$(U^*(b) < \gamma) \wedge (A(b) \geq \alpha_L) \wedge (A(b) < \alpha_H \vee V(b) > \beta)$$

Underutilized: although addresses are occasionally used, they show low $A(b)$ values.

$$A(b) < \alpha_L$$

Unclassifiable: we decline to classify blocks with few active responders, currently defined as any block where fewer than 20% of addresses respond.

We selected these categories to split the majority of the $(A(b), V(b), U^*(b))$ space, informed by evaluations of dozens of blocks (573K addresses in total) backed by manual probing of hosts and hostnames (for details about categories, see Appendix C, and about the $(A(b), V(b), U^*(b))$ space, Appendix B).

While we have defined these categories based on what we can observe, the categories are correlated to real-world address usage. *Always-stable* is typical of servers, routers and always-up end hosts. Manual inspection of randomly chosen reverse hostnames indicates that more than 80% servers and routers have always-stable addresses. *Sometimes-stable* correlates addresses with hostnames that indicate statically-assigned user computers, businesses (names containing “biz” or “business”), some dynamically assigned but always-on connections (cable modems or DSL connections). *Intermittent* characterizes the majority of cable and DSL hosts and some active dial-up hosts. We find many address blocks, often identified as dial-up by hostname, are categorized as *underutilized.* (More than 50% of hostnames that indicate dial-up have $A(b) < \alpha_L$.)

Appendix C relating these ping-observable categories to several hostname-inferred usage categories which represent real-world address usage.

We examine sensitivity to our choices in Section 5.3. Appendix B shows how these terms divide the space.

3.5 Identifying Low-bitrate Blocks

Block categories correlate with edge link technologies, but they are not one-to-one—we find that dial-up and DSL appear as both intermittent and underutilized. To better understand technology trends, we next show that variance across repeated RTT measurements can identify blocks with low-bitrate edge links. We define low-bitrate as less than 100Kb/s, such as dial-up (56Kb/s) and GPRS (57.6 Kb/s). We first present a RTT model, and then apply it.

3.5.1 *Background: components of RTT*

Round-trip time has several components:

$$RTT = 2(D_{cpu} + D_{prop} + D_t + D_q)$$
$$\text{where } D_t = S/B \text{ and } D_q = nD_t$$

The first two components, per-hop processing delay in the routers (D_{cpu}), and distance-based propagation delay (D_{prop}) are largely independent of the edge link. Transmission delay (D_t), however, is based on packet size (S, approximated as constant for this simple model) and the bottleneck link's bitrate, B. Queuing delay (D_q) is a multiple of D_t based on queue length. (All terms are for the full round-trip and do not require path symmetry; we assume the prober is well connected.)

Our goal is to distinguish addresses with low-bitrate edge links from broadband links. In the simplest possible case, we first assume the targets are one-hop from our prober and there is no congestion, so D_{cpu} and D_{prop} are negligible and $D_q = 0$. Here the only difference is transmission delay, and we can easily distinguish common edge technologies (Table 2) since D_t dominates RTT. Here even a simple threshold of $R^*(a)$ would distinguish slow edge links, since our 64B probe takes 9ms over a 56kb/s dial-up link but much less than 1ms at broadband (1Mb/s or faster).

In practice, our prober is distant from most of the Internet and we encounter interfering traffic. At long distances, D_{cpu} and D_{prop} can dominate RTT, often approaching 200ms for communications between continents, completely obscuring the effects of the edge we wish to observe via D_t.

Queuing delay is another source of noise, but it also provides the *means to see through* distance. With queuing delay, $D_q = nD_t = n(S/B)$, where B is the bitrate on the backlogged link. Queuing delay can happen at any location along the path, either in the backbone or the edge link. We assume that most queuing occurs at the edge link, since although backbones are highly multiplexed, they consist of high-bitrate, carefully managed links, and we expect queues to be short (n is low) and to clear quickly (since $D_t < 1\mu$s at 1Gb/s, even for a 1500B interfering packet). For slow edges, each packet in the queue ahead of a probe adds tens or hundreds of milliseconds, since (D_t is 1ms for 1Mb/s ADSL, and almost 10ms for dial-up, and $D_q = nD_t$. If we assume slow links are likely at the edges, then queuing (D_q) and RTT are dominated by the effects of this edge link.

link type	**transmission delay**	
packet size	64B	1500B
56Kb/s dial-up	9ms	212ms
1Mb/s ADSL	0.5ms	12ms
1Gb/s Ethernet	0.5us	12us

Table 2: Common link type and the transmission delay for 64KB and 1500KB packet respectively

3.5.2 *Identifying low-bitrate links from RTT*

We next turn to identifying blocks with low-bitrate edge-links with three steps: isolating the D_q component of RTT, and generalizing results to blocks, and then classifying blocks as low-bitrate.

Any given RTT observation is made up of the four components identified previously. With one observation we cannot separate those contributions. However, a week-long survey provides hundreds of observations for most addresses. If routing is generally stable, all components of RTT are constant except for queuing delay, while D_q varies depending on how backlogged the edge link each time it is probed. We therefore look at *variation* in RTT to infer D_q, as measured by $R^*_\sigma(a)$, the standard deviation of the RTT. Routing techniques such as load balancing or wide geographic distribution of adjacent addresses [12] are sources of noise; we utilize a fairly high threshold to mitigate their effects.

Standard deviation is well defined only with multiple measurements and for positive probe responses; we ignore $R^*_\sigma(a)$ when $|R^*(a)| < 10$ as statistically invalid, and RTTs for negative responses since they may be generated by a router on either side of the edge link. There are many addresses that fail to reply positively to probes: in our survey, only about 41% of addresses from blocks that have any responses at all respond, and about one-twentieth of these respond fewer than 10 times. Our analysis of networks shows that most are composed of large, homogeneous blocks (we show this data in Section 4.1), so we extend our address-level observations to blocks by defining a block-level estimate of RTT variance as the median of all address-level standard deviations: $R^*_{\mu_{1/2},\sigma}(b) = \text{median}(R^*_\sigma(a))\ \forall a \in b$.

Low-bitrate block: We therefore identify low-bitrate

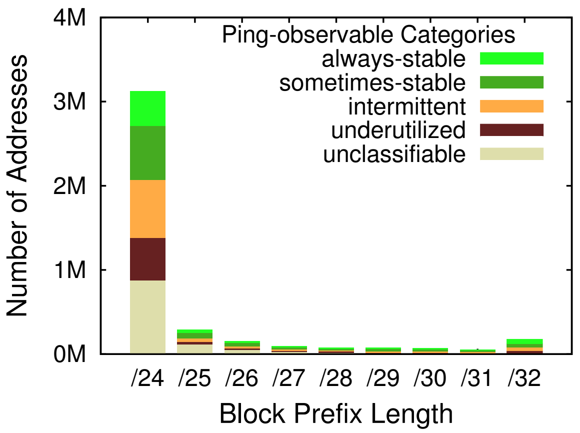

Figure 2: Number of addresses in each block size and ping-observable categories in *IT17ws*.

blocks from broadband by large variance:

$$R^*_{\mu_{1/2},\sigma}(b) > \delta$$

We select $\delta = 300ms$, because it is roughly 1.5× the delay of a full-size packet at dial-up speeds (1500B takes 212ms at 56kb/s), and based on evaluation of dozens of low-bitrate blocks. We examine the validity of this classification approach and the threshold in Section 5.

4. APPLICATIONS

We next use the data to explore several questions in network management: what are typical sizes of consistently used Internet address blocks? How effectively are they being used? And how prominent is dynamic addressing?

To help answer these questions we compare our observations with the allocation data from the regional Internet registries (RIRs) [1]. This RIR data includes the time and country to which each address block is assigned. Although not completely authoritative, this data is the best public estimate for address delegation of which we are aware. We collect data from each of the RIRs, selecting data dated June 13, 2007 to closely match our survey data.

4.1 Block Sizes

We begin by considering block sizes. Figure 2 and Table 3 show our analysis of *IT17wvs*.

This data shows that addresses in the Internet are most commonly managed in blocks with /24 prefixes. In fact, even though there are more opportunities for small blocks, we find more /24 blocks than blocks of size /25 through /29. Since our data collection only probes consecutive runs of 256 addresses, this prevalence suggests we may need to probe larger consecutive areas to understand if even larger blocks are common but not seen in our survey.

There are a very large number of the smallest blocks, with about as many /29s as /24s, and roughly twice as many /30s as /29s, and /31s as /30s. These results may be artifacts of our block discovery algorithm: it is statistically easier for an address to be consistent with a few neighbors in a small block than with 128 neighbors in a /25. We next re-examine the second assumption underlying our work: are contiguous addresses often used similarly? If we define consistent usage as just the largest three block sizes (/24 through /26) that we successfully identify, we find 2,529,216 addresses are used consistently, or 44% of the probed address space.

While clearly defined, this percentage does not accurately present how much of the Internet is consistently used. Some of the probed address space is unclassifiable (with consistent usage but fewer than 20% of addresses responding), or completely non-responsive. We cannot say anything about blocks that fail to respond at all. The status of unclassifiable blocks is uncertain, but a conservative position is to declare them inconsistent. A more representative evaluation of the Internet is therefore to compare how much is definitely used consistently (2.5M addresses in large blocks) against that is effectively inconsistent (the 506,178 addresses in small blocks) and the possibly inconsistent (the 1,087,472 addresses in unclassifiable blocks). This computation suggests that a lower bound of *61% of the responsive Internet is used consistently*, We believe this supports our second assumption: **the majority of contiguous addresses are used consistently**.

4.2 Address Utilization

Given block sizes, we next evaluate how efficiently addresses are used in those blocks. Inefficient IPv4 usage represents an opportunity for improvement, but greater efficiency comes with greater management cost. Management cost of IPv4 should be weighed against simpler-to-manage IPv6.

4.2.1 Quantifying underutilization and possible causes

The *underutilized* ping-observable category is defined as a sequence of addresses that are used less than 10% of the time (Section 3.4). Large blocks of such infrequently used, public IP addresses generally indicate inefficient address utilization. (Such low utilization seems to make sense only in unusual circumstances, such as a DTN satellite only infrequently in view [9].)

The underutilized column of Table 3 shows that these blocks are quite common, accounting for 17–23% of blocks of each size, Although not shown in the table, the mean availability of addresses in /24 underutilized blocks is only 3.2% of our 10-day observation (*IT17ws*). Manual examination of addresses shows the mean number of up periods is less than 5 ($V^*(b) = 4.6$), typically for around 1 hour ($U^*(b)$).

size			sometimes-			classifiable	unclassifiable	blocks	addresses
pfx	addrs	always-stable	stable	intermittent	underutilized	(100%)		*[100%]*	
/24	256	1,603 (18%)	2,517 (29%)	2,673 (30%)	1,994 (23%)	8,787*	3,411 *[27%]*	12,198	3,122,688
/25	128	323 (23%)	523 (38%)	295 (21%)	237 (17%)	1,378*	920 *[40%]*	2,298	294,144
/26	64	346 (21%)	617 (38%)	378 (23%)	274 (17%)	1,615*	787 *[33%]*	2,402	153,728
/27	32	432 (20%)	855 (40%)	506 (23%)	361 (16%)	2,154†	872 *[29%]*	3,026	96,832
/28	16	759 (20%)	1,301 (34%)	993 (46%)	734 (19%)	3,787†	1,139 *[23%]*	4,926	78,816
/29	8	2,077 (21%)	3,190 (32%)	2,355 (24%)	2,227 (23%)	9,849†	0	9,849	78,792
/30	4	3,312 (19%)	5,656 (33%)	4,679 (27%)	3,707 (21%)	17,354†	0	17,354	69,416
/31	2	4,195 (16%)	9,867 (37%)	7,864 (30%)	4,566 (17%)	26,492†	0	26,492	52,984
/32	1	52,646 (30%)	42,847 (24%)	43,266 (25%)	36,707 (21%)	175,466†	0	175,466	
entire *IT17ws* dataset:		(1,603,086 addrs. in non-responsive blocks) + (4,122,866 in responsive blocks)						22,367	5,725,952

Table 3: Number of blocks of each size in *IT17ws* (10 days). Unclassifiable percentages relative to all blocks; other percentages relative to classifiable blocks. Asterisks: consistent blocks, daggers: non-consistent.

To understand causes of underutilized blocks we examine the address hostnames of these /24 blocks. We find 63% of addresses provide hostnames, and many of these hostnames (34%) include keywords that suggest how the address is used. For example, dial and dsl suggest edge link technologies, and dynamic or pool suggest dynamic address assignment. (Full details are in Appendix C.2.) Among the various usage suggested by hostnames, underutilized blocks are correlated with *pool* (68%), *ppp* (56%) and *dial* (54%) hostname categories.

We hypothesize that this low utilization is tied to dial-up technology itself. Dial-up lines are often shared with voice communication, encouraging short, intermittent use. Yet dial-up POPs must be provisioned to handle peak loads. A secondary factor may be trends shifting customers from dial-up to higher speed connections. Perhaps old dial-up provisioned blocks are simply in lower demand than previously. Finally, while dial-up utilization is low, we cannot tell how many users each dial-up address serves. Perhaps address reuse is high enough to make these apparently under-provisioned addresses a bargain relative to supporting the same number of users with always-on connections. Further study to understand these trade-offs is future work.

Reversing the question, we can ask *which blocks are well utilized*? Still by examining the hostnames, we found that blocks with keywords *static, cable, biz, res, server, router* have very few underutilized addresses. Static addresses are usually assigned to fixed-location desktops or businesses, and these computers tend to maintain Internet connection and occupy their address for a fairly long time. In addition, static addresses are often billed at a flat rate per month, while dynamic addresses may incur a time-metered charge.

4.2.2 Locations and trends of underutilization

Evaluating underutilization by country may highlight policy differences by regional registries or ISPs. After merging our data with RIR data, Table 4 shows utilization by country. We see that the United Kingdom and Japan have the largest fraction of underutilized blocks, 40–60%, suggesting potential local policy differences. We expected a large number of underutilized blocks in the U.S. because of wide deployment of dial-up. While the U.S. has the largest absolute number of underutilized blocks, its fraction is relatively low.

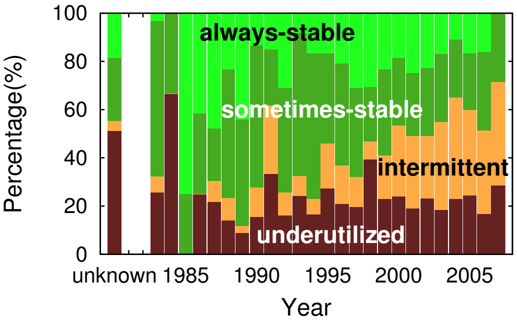

Figure 3: Trend of ping-observable category change in *IT17ws* /24 blocks

Table 5 shows that the fraction of underutilized blocks is fairly consistent across all five RIRs, suggesting differences are likely due to country, not RIR policies.

Finally, the lower right graph in Figure 3 shows when underutilized blocks were allocated. The fraction of blocks by age seems fairly evenly distributed, except for peaks in very early allocations (1984 and unknown), where more than 60% of the blocks assigned are underutilized. We believe these earliest allocations were made with relatively little assessment of organizational need, and large initial allocations allow continued use with minimal concern for efficiency.

4.3 Intermittent and Dynamic IP Addressing

Addresses are intermittently used by statically addressed hosts that are only sometimes connected to the network, or by hosts that obtain dynamically assigned addresses from a pool, typically with DHCP [7].

Dynamic assignment of addresses allows ISPs to multiplex many users over fewer addresses. Dynamic addressing also provides ISPs the business opportunity of offering static addresses as a higher-priced service, and

code	country	always-stable	sometimes- stable	intermittent	underutilized	classifiable (100%)	unclassifiable	blocks *[100%]*
US	US	673 (27%)	**1,106 (45%)***	231 (9.3%)	472 (19%)	2,482	1,383 *[36%]*	3,865
CN	China	39 (4.1%)	117 (12%)	**615 (65%)***	171 (18%)	942	132 *[12%]*	1,074
JP	Japan	**383 (48%)***	50 (6.2%)	18 (2.2%)	**350 (44%)***	801	288 *[26%]*	1,089
DE	Germany	65 (10%)	125 (20%)	**388 (61%)***	62 (9.7%)	640	56 *[8.0%]*	696
KR	Korea	21 (4.6%)	131 (29%)	**237 (52%)***	68 (15%)	457	142 *[24%]*	599
FR	France	18 (4.1%)	**227 (52%)***	167 (38%)	28 (6.4%)	440	58 *[12%]*	498
GB	UK	39 (13%)	37 (12%)	52 (17%)	**179 (58%)***	307	180 *[37%]*	487
BR	Brazil	7 (3.9%)	35 (19%)	**86 (48%)***	52 (29%)	180	58 *[24%]*	238
all others		358 (14%)	689 (27%)	879 (35%)	612 (24%)	2,538	1,114 *[31%]*	3,652
/24 blocks in entire *IT17ws* dataset:						8,787	3,411 *[27%]*	12,198

Table 4: The distribution of /24 blocks in ping-observable categories of 10 countries. Bold and asterisks indicate the categories with more than 40% of blocks. Colors indicate categories and each country's dominant category. Countries are sorted by total number of blocks.

registry	always-stable	sometimes- stable	intermittent	underutilized	classifiable (100%)	unclassifiable	blocks *[100%]*
RIPENCC	408 (14%)	798 (27%)	**1,084 (37%)***	661 (22%)	2,951	990 *[25%]*	3,941
APNIC	473 (18%)	422 (16%)	**1,091 (40%)***	716 (27%)	2,702	795 *[23%]*	3,497
ARIN	706 (27%)	**1,185 (45%)***	258 (9.7%)	512 (19%)	2,661	1,481 *[36%]*	4,142
LACNIC	13 (3.2%)	94 (23%)	**218 (53%)***	86 (21%)	411	120 *[23%]*	531
AFRINIC	3 (4.9%)	18 (30%)	**21 (34%)***	19 (31%)	61	19 *[24%]*	80
/24 blocks in entire *IT17ws* dataset:					8,787	3,411 *[27%]*	12,198

Table 5: The distribution of /24 blocks in ping-observable categories of 5 regional registries. Bold and asterisks indicate the categories with more than 40% of blocks. Colors indicate categories and each registry's dominant category. Registries are sorted by total number of blocks.

potentially makes it more difficult for users to operate servers. Dynamic addressing has been promoted to users as a security advantage, on the theory that a compromised computer is more difficult to contact if its IP address changes. Dynamic addressing prevents users from running services or accepting unsolicited inbound connections (for example, for incoming SIP calls), although applications employ work-arounds such as STUN [30].

Recent studies [15,34,40] have examined dynamic addressing for several reasons. First, dynamic addresses complicate some network services, such as reputation systems. They also are correlated with spam; some spam filters penalize dynamic addresses because of the frequent exploitation of dynamically addressed home computers by spammers. We next show that our approach can identify dynamic addressees and suggest the causes and trends that have been previously invisible.

4.3.1 Quantifying dynamic addressing

We believe that the *intermittent* and *underutilized* ping-observable categories correspond to the short-term dynamically assigned addresses of interest. Although we cannot quantify what fraction of these categories actually use DHCP, our belief is supported by hostname analysis. Hostnames shows that intermittent blocks commonly include keywords *cable* (57%), *dynamic* (48%) and *dsl* (41%), all of which often use short- or moderate-term dynamic addressing, and underutilized blocks often include keywords for *pool* (68%), *ppp* (56%) and *dial* (54%).

Table 3 shows that 40–50% of classifiable blocks (depending on block size) appear to be dynamic. Even with wide deployment of always-on connectivity, nearly half of Internet addresses are used for short periods of time. For intermittent blocks, the mean availability is just under 30%, with nine use periods over the week and a mean U^* around 2.5 hours.

4.3.2 Locations and trends for dynamic addressing

Analysis by country can suggest how political, cultural and policy factors affect addressing. Table 4 shows that nearly two-thirds of Chinese blocks are intermittent, with Germany, Korean, and Brazil all nearly half or more. Several factors may contribute to this use.

China has a very large population and is a relative latecomer to the Internet; from the beginning of commercial deployment in China. ISPs have planned to make best use of the relatively few IPv4 addresses per potential user. They have therefore promoted dynamic use to improve address utilization. An interesting direction for future work would be to evaluate how effective their utilization is. Unfortunately we only know address responsiveness, not the number of actual computers users per address needed to answer this question.

Time-metered billing is another reason for intermittent use. Parts of China and Germany employ metered billing, encouraging intermittent use even with broadband. Other potential reasons for intermittent use include turning off a router to conserve energy, or carrying over habits learned from dial-up use to broadband, and potentially continued use of dial-up connections shared with voice communication.

Evaluation of usage by registry (Table 5) shows larger differences in use. We see that intermittent blocks are very prominent under APNIC and LACNIC (40–53%), five times more common than for ARIN in North America (9%). We believe these differences stem largely from policies of the countries the RIRs serve, not the RIRs themselves. We discussed Chinese practice above; several Latin American countries have limited choice in ISPs, with national providers adopting pricing or policies that strongly favor dynamic address assignment even for business use (as confirmed by LACNIC personnel [23]). We speculate that the large number of sometimes-stable blocks in ARIN is because of long DHCP lease times and always-on use by home users, enabled by relatively plentiful numbers of IPv4 addresses per user.

Finally we consider trends in dynamic addressing. The lower left of Figure 3 shows that intermittent blocks are more common in new address allocations. This observation is consistent with a recognition of eventual full allocation of the IPv4 address space and efforts to manage addresses in countries newer to the Internet. The rise in intermittent blocks matches a corresponding fall in always-stable blocks (top left, Figure 3). In addition to growing demand for dynamic addressing, this trend suggests most new addresses are added to provide service for home users, intermittently. While the absolute numbers of always-stable businesses and servers grows, its fraction of all addresses is shrinking.

4.4 Understanding Edge Bitrates

To understand causes for utilization, we next look at block connectivity to the Internet.

In Section 3.5.2 we suggested that RTT variance can indicate low-bitrate edge links such as dial-up and pre-3G mobile telephones. Here we apply this analysis to provide a new tool to understand how edge networks correlate with underutilization. Future work includes using this analysis to evaluate deployment trends and to automatically adapt websites to the user's network.

To understand the usage of low-bitrate blocks, Figure 4 shows the availability for blocks broken into low- and non-low-bitrate groups by RTT stability (as defined in Section 3.5.2). From the underutilization threshold of $A(b) < 0.1$, we see that nearly 80% of low-bitrate blocks are underutilized, compared to only 20% of non-low-bitrate blocks. Therefore low-bitrate connections strongly correlate with sparse use.

To explain this correlation between edges and underutilization, we use hostnames and whois to infer *operational usage*—roughly, how blocks are managed (dynamic or static) and what type of edge-link they are (dial-up, PPP, DSL, etc.). Such inferences are less than ideal, but they provide the best available ground truth about the general Internet. Among the 200 randomly selected low-bitrate blocks, we successfully inferred the operational usage of 46 blocks: 41 dial-up, 2 PPP, and 3 DSL blocks. Dial-up and PPP are indicator of low-bitrate edge connection while DSL is one representative of broadband connection. While not providing definitive causes of underutilization, this suggests correlation between low use rates, low bitrates, and dial-up edge networks.

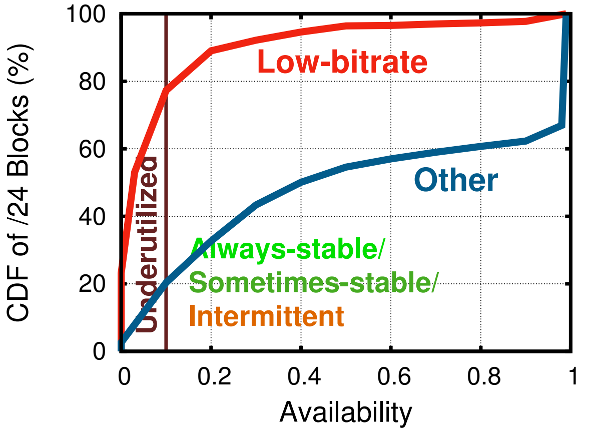

Figure 4: Comparison of availability for low-bitrate (top line) and non-low-bitrate (bottom line) classifiable /24 blocks in *IT17ws*.

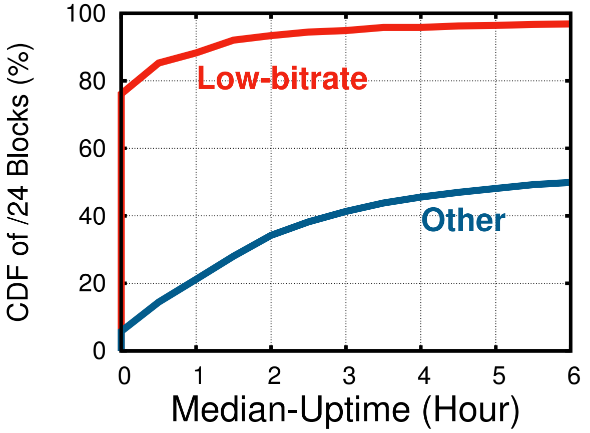

Figure 5: Comparison of Median-Up between Low-bitrate with Non-low-bitrate Classifiable /24 Blocks in *IT17ws*.

To support this explanation, we studied the median-uptime U^* for both low-bitrate and broadband blocks shown in Figure 5. We found that up durations in the vast majority of low-bitrate blocks are quite brief: 85% of low-bitrate blocks have a $U^*(b) < 0.5$ hours, compared to only 15% of other blocks. This observation suggests that low-bitrate, dial-up blocks are provisioned for a large number of potential users who do not use the network concurrently. We hypothesize two reasons for this, 1) low-bitrate edge connection lead to long delay, which lead to less satisfying user experience, which lead to shorter occupation time; 2) the billing of low-bitrate edge connection are usually usage-based instead of flat-rate, thus it is natural for the users to disconnect from network soon after their task is done.

5. VALIDATION

We have now shown data to support our three assumptions: addresses respond to probes (the subject of prior work [15]), adjacent addresses have similar use (Section 4.1), and probes suggest use (Section 4.2). These results have two limitations, however. First, since they are based on active probing, they are only available for the portion of the Internet that respond to probes. Evidence suggests that somewhat more than half of the publicly addressed hosts respond [15]; extension of these results to the whole Internet is an area of continuing work. Second, our conclusions are based on data taken from one survey (*IT17ws*) from the general Internet. While not biased, we cannot compare these results to the true network configuration that is distributed across thousands of enterprises.

We next present three additional studies to further validate these assumptions and address the second limitation. First we evaluate data taken from USC, a smaller and potentially biased dataset, but one where we have ground truth from the network operations staff. We then extract small random subsets of the general Internet and infer the ground truth by manual inspection using ISC-DS hostname data [17] and the whois database. Finally, we compare our Internet-wide results with additional data taken one-half to two years later to verify that our conclusions do not reflect something unusual in a single measurement or time.

5.1 Validation within USC

We first compare our methodology against ground truth obtained directly from the network operators at USC. This section uses dataset *USCs* and applies the same analysis used on our general Internet dataset.

Figure 6 shows block sizes and classifications from our approach.

Block identification and classification at USC shows a similar prevalence of /24 blocks (85% of USC addresses are in /24s, compared to 61% in the Internet). However, USC shows many fewer intermittent and underutilized blocks compared to the Internet (only 8% among classifiable /24 blocks, Figure 2); we expect such variation across enterprises. We next use this data to evaluate how our assumptions affect our ability to accurately find block size, consistency, and usage.

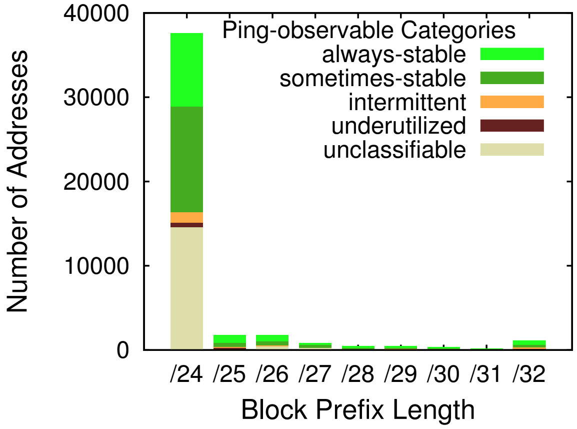

Figure 6: Number of addresses in each block size and ping-observable categories in *USCs*.

category:	blocks	percentage	
in routing table	243	100%	
false negative	105	43%	
not in use	19		
not responding	28		
few responding	12		
single-block multi-usage	46		
/25 to /27	9		
/28 to /32	37		
blocks identified	147		100%
correctly identified	138	57%	94%
false positive	9		6.1%
multi-block single-usage	9		

Table 6: Evaluation of accuracy of block identification USC to ground truth sizes.

5.1.1 *Validation of block identification and sizes*

To validate our estimation of block sizes, we compare our analysis with the internal routing table from our network administrators. This data helps quantify the accuracy of our approach, measuring the *false positive rate*, blocks that we detect but that do not actually exist, and the *false negative rate*, blocks that exist but we fail to detect.

Table 6 summarizes our comparison for all /24 blocks. (Smaller blocks are not present in our ground-truth routing table.) We find our approach correctly identifies 57% of all blocks in ground truth. Although we find the majority of blocks, we have a significant number of false negatives, failures to detect blocks. For this dataset, these false negatives show *our approach is somewhat incomplete.* On the other hand, if we evaluate our algorithm by what it says, we see very few false positives, correctly identifying 94% of all blocks we detect. For this dataset, almost no false positives show *our approach is quite accurate in what it asserts.*

To understand accuracy, we looked at when our approach incorrectly identifies blocks. All nine false positives are due to multiple blocks with common usage.

category:	blocks	percentage
classified	138	100%
unclassifiable (*false negative*)	52	38%
incorrectly classified (*false positive*)	3	2.1%
always stable (dynamic)	3	
correctly classified (*true positive*)	83	60%
intermittent (dynamic)	4	
sometimes stable (dynamic)	5	
intermittent (VPN)	1	
underutilized (VPN/PPP)	2	
always stable (lab)	2	
sometimes stable (lab)	2	
always stable (building)	25	
sometimes stable (building)	42	

Table 7: Evaluation of block classification accuracy at USC to ground truth.

We examined each incorrect block and found that USC administrators had placed two logically different blocks on adjacent addresses, but these administratively different blocks were used for similar purposes. Since our evaluation is based on external observations of use, we believe there is no way *any* external observer could determine these administrative distinctions.

For false negatives, we found several sources of missed block identification. We found that many blocks were either in the routing table but not assigned to locations or services (19 not in use), or in the routing table and assigned, but with no ping responses (28 not responding), or filled with only a few responders (12 few responding). In each case, our algorithm refuses to make usage assertions on unused or sparsely used space. Non- or few-responding blocks may be due to firewalls, reflecting a limitation of our probing method. Not-in-use blocks would be impossible for any external observer to confirm. In principal our algorithm could identify non-responsive blocks, but it is difficult for external observation to distinguish unused from firewalled space.

Finally, other false negatives occur due to blocks that have been administratively assigned as /24s but then are used for different purposes. Nine of these show large, consistent patterns, possibly indicating delegation at the department level that is not visible to university-wide network administrators. If so, these represent incompleteness in our ground-truth data. Smaller mixed-use blocks represent violations of our assertion that adjacent addresses are used consistently.

5.1.2 *Validation of block classification and usage*

Table 7 shows the accuracy of our approach for the 138 blocks we classify. We declare 38% unclassifiable (false negatives); here we have discovered the correct block size but decline to declare a ping-observable category because the block is only sparsely responsive. We correctly classify the majority of blocks, selecting ping-observable categories that are consistent with the use of 60% of blocks. We mis-identify three blocks (a 2% false positive rate), all reported as dynamically allocated but observed as always stable. These blocks perhaps represent DHCP-assigned addresses with very long lease times for computers that are always up.

5.1.3 *Validation of edge bitrate*

We also validated our edge-bitrate assessment. USC has only two low-bitrate blocks (dial-up blocks running PPP). Experimental evaluation of *LTUSCs* successfully identifies both as low-bitrate, and does not mis-identify any of the 136 other blocks as low-bitrate. While this 100% accuracy is reassuring, the proximity of prober and target suggests that our validation with random Internet blocks (Section 5.2.3) is a more general result.

5.2 Validation in the General Internet

Our main validation results use USC because there network operations can provide ground truth. We would like to evaluate how well our approach works on the general Internet as well, since commercial use may differ from USC. We evaluate our ping-observable classification results for 100 randomly selected /24 blocks, and enlarge the sample size for our edge-bitrate validation in Section 5.2.3.

While we cannot get ground truth from network operations for the general Internet, we can get clues about block size and usage from hostnames and the *whois* database. Hostnames are often assigned in patterns that suggest common administration and access method. For example, hostnames in 4.168.174/24 follow the convention `dialup-4.168.174.*.dial1.losangeles1.level3.net`. Such consistent naming conventions strongly suggest a common administrator (in this case, Level 3). Second, the presence of "dial" in the name suggests dial-up usage and low-bitrate connection. Whois information provides an alternative view. For example, hostnames in 70.204.31/24 follow the convention `*.sub-70-204-31.myvzw.com`. Names suggest common administration, but not how it is used. Whois indicates this block is assigned to *Cellco Partnership DBA Verizon Wireless*, suggesting mobile phone usage.

In the 100 /24 blocks, 47 of them are not found in ISC dataset, 7 of them have too few hostnames. Because we can not gain the ground truth from these blocks, we exclude them from our validation. In the 46 /24 blocks left, 37 of them are identified as /24 by hostnames, 7 of them are potential /24 blocks (hostnames do not follow exactly the same convention but share similar keywords), 1 block is inferred to be a /25 block, and 1 block is inferred to be composed of 1 /30 and 2 /32 blocks. Because most addresses are in /24 blocks, to simplify, we just give the statistics of the 37 identified by hostnames as /24 blocks and 7 as potential /24 blocks in Table 8. The /25 is not successfully identified due to

category:	blocks	percentage	
/24 randomly selected	100	100%	
decided (/24 inferred from hostname)	37	37%	100%
correct	25		68%
wrong (false negative)	12		32%
few responding	6		
single-block multi-usage	6		
undecided	63	63%	
no hostname	45		
few hostnames	7		
potential /24 inferred	7		
correct	7		
has sub-/24 groupings	4		

Table 8: Evaluation of block identification accuracy of random Internet blocks.

category:	blocks	percentage
classified	20	100%
unclassifiable (*false negative*)	2	10%
incorrectly classified (*false positive*)	1	5%
sometimes-stable (server)	1(1)	
correctly classified (*true positive*)	17	85%
always-stable (server, biz)	3(2, 1)	
sometimes-stable (dsl, static)	3(2, 1)	
intermittent (dsl, cable)	4(3, 1)	
intermittent (mobile, dynamic, dial)	4(2 ,1, 1)	
underutilized (pool-pond, dsl, client)	3(1, 1, 1)	

Table 9: Evaluation of block classification accuracy of commercial blocks

too few responding, the 1 /30 and 2 /32 are correctly identified as small server blocks.

5.2.1 Validation of block identification and sizes

We randomly select 100 /24 blocks probed, and compare their clustering results with our best estimates about the ground truth from manual analysis of hostname and whois in Table 8 (37 are identified as /24 by hostnames).

As shown in Table 8, the correctly identified rate (68%) is even higher than the one in USC validation (57%). The reason is that address space in the general Internet is used in a bigger granularity than campus network, thus, blocks tend to be more consistent.

Although the results seem not bad, most blocks are /24, we would like looking deeper into see how our method works on smaller blocks. We randomly selected 20 /24 blocks identified as composition of smaller blocks. 4 of them have too few hostnames, 12 are actually /24 blocks but break into smaller blocks due to non-consistent usages. 2 are partially responding, i.e., only half of the block is responding, the other half is probably firewalled. 2 /25s are correctly identified. This fact suggests that most smaller blocks we found are potentially big blocks broke into pieces due to non-consistent usage, however, we did find some consistent small blocks identified by hostnames.

5.2.2 Validation of block classification and usage

To validate the ping-observable classification, we look at the 25 correctly identified /24 blocks in the previous 100 random /24 blocks. To validate the low-bitrate classification, because of the low percentage of low-bitrate blocks, we enlarged our random sample to 200 /24s.

About ping-observable classification, of the 25 correctly identified /24 blocks, we classified 20 of them; 5 were unclassifiable because of lack of hostname and whois information. We summarize the validation results in Table 9. Because we have a loose mapping between ping-observable category and hostname-inferred usage category, the correct rate is relatively high (85%). For example, 2 DSL blocks are sometimes-stable, 3 DSL blocks are intermittent and 1 DSL block is underutilized. We consider the above three mappings being all correct. Although one might argue that host firewalls could lower down the availability metric which leads to the classification of underutilized blocks. The results show that most DSL blocks are still distributed among intermittent and sometimes-stable categories which weaken the claim of host firewall popularity. Because the mapping from hostname-to-ping-inferred category is not one-to-one, our estimates of "ground truth" here are imprecise and we do not claim this result is definitive, but merely suggestive that our classification works well over the general Internet.

5.2.3 Validation of edge bitrate

Among the random 100 /24 blocks, only 10 of them (6 dsl & 1 cable, 1 dial & 2 mobile-phone) can be used as ground truth to validate our edge-bitrate assessment. Known low-bitrate blocks are rare in the Internet, thus we want to have more samples to validate our edge-bitrate assessment. Simply adding more random blocks to the previous 100 blocks and manually inspecting them is time-consuming. So we use an automatic way, although a little coarser, to add more samples. We randomly pick only classifiable /24 blocks with consistent naming convention in hostnames that have certain keywords (dsl, cable, dial) indicating edge access link type. This process can be easily automated with hostname data only without querying the whois database. Thus, in addition to the previously identified 10 blocks, we add 26 random hostname-inferrable edges blocks, for a total of 36 blocks as ground truth. Table 10 summarizes our analysis.

For the 36 blocks where we can infer edge types to evaluate accuracy, we successfully classify all low-bitrate blocks and all broadband blocks. Our low-bitrate detection algorithm provides an 0% false-negative rate and a 0% false-positive rate. There were three confusing-hostname broadband blocks classified into low-bitrate. These blocks have *dial* in their hostnames. However, when we confirmed with the ISP's network operations, these blocks are actually fast 3G/UMTS wireless con-

category:	blocks	percentage
hostname-inferrable edges	36	100%
low-bitrate blocks (6 dial, 2 mobile)	8	
$R^*_{\mu_{1/2},\sigma}(b) > \delta$ (true positive)	8	
$R^*_{\mu_{1/2},\sigma}(b) \leq \delta$ (false negative)	0	0%
broadband (21 dsl, 4 cable, 3 3G)	28	
$R^*_{\mu_{1/2},\sigma}(b) > \delta$ (false positive)	0	0%
$R^*_{\mu_{1/2},\sigma}(b) \leq \delta$ (true negative)	28	
clear hostname	25	
confusing hostname	3	

Table 10: Evaluation of low-bitrate block classification accuracy of commercial blocks.

nections. Their R^*_σ values are 20ms, 41ms and 43ms respectively, suggesting 1Mb/s links or faster.

5.3 Consistency Across Repeated Surveys

We next wish to understand if the parameters of our data collection or analysis have a disproportionate effect on our conclusions about Internet-wide address usage. To do so, we compare analysis of *IT17ws* with that of three new datasets, *IT16ws*, taken five months earlier; *IT30ws* and *IT31ws*, taken 30 months later. These surveys allow us to consider both adjacent surveys at two different times, and longer-term trends. Half of the /24 blocks in the survey are consistent across each survey, and half are randomly chosen in each survey (full details of selection methodology are elsewhere [15]). This comparison therefore observes whether network changes alter observations of the same blocks, and whether different sets of blocks show very different behavior.

Our estimates of the block size distributions are almost identical in the four surveys. If we define s_p as the vector of number of blocks of prefix length p, the correlation coefficient of the vectors for *IT17ws* against all other surveys are all above 0.9989. We conclude that a random sample of 1% of the Internet is large enough that the block size observations are hardly affected if half of the sample is changed.

Our work assumes that contiguous addresses are often used consistently. Following Section 4.1, we consider blocks of size /24 through /26 as consistent, and size /27 through /32 as inconsistent. These percentages are quite consistent in adjacent surveys, with a possible slow downward trend over time: In *IT17ws*, 44% of probed Internet, going to 43% in *IT16ws*, and 38% later in both *IT30ws* and *IT31ws*. Results are similar if we consider percentage of the responsive Internet, with 60% and 61% in *IT16ws* and *IT17ws*, and 57% and 58% in the later two surveys.

Finally we consider the temporal consistency of our ping-observable classification across four surveys. We show that temporarily adjacent surveys show consistent classification results, while more distant surveys show greater divergence. First we compare each adjacent pair of surveys. For *IT17ws* and *IT16ws*, initially we found the correlation of the number of blocks in each category to be generally good but not great across all block sizes—it ranged from 0.663 to 0.938 for blocks smaller than /29, but the correlation for /24 blocks was only 0.349. Examination showed that around 500 blocks were shifting between always- and sometimes-stable. This shift occurred because of a change in volatility and our selection of the always-stable requirement that $V(b) \leq \beta$ and $\beta = 0.0016$. For very stable hosts, a few outages can change $V^*(b)$ significantly. Examination showed that *IT16ws* and *IT17ws* are of different duration (6 and 10 days). A longer duration makes it easier to distinguish between sometimes- and always-stable blocks. When we keep the observation duration constant by considering only a 6-day subset of *IT17ws*, the correlation coefficient for /24 classification rises to 0.626. We conclude that most ping-observable classifications are good, but the separation between sometimes- and always-stable categories is somewhat sensitive. We plan to investigate the sensitivity in future work by down-sampling the survey data in time. We confirm this result by comparing the later surveys, where full 14-days of *IT30ws* and *IT31ws* show the correlations ranging from 0.77338 to 0.987. Thus we conclude that results taken near the same time are fairly consistent.

We next compare surveys taken two years apart: *IT17ws* and a 6-day subset of *IT31ws* (data for *IT31ws* is in Appendix A). There are two main differences: first, many blocks shift from sometimes-stable to always-stable, where *IT31ws* has 2,459 always-stable /24 blocks compared to only 2,001 before (33% vs. 23%). The percentage of intermittent and underutilized blocks is similar. Second, we see a larger number of /32 blocks in the later survey, up to 198k from 179k, with many of the new blocks shown as /32 always-stable blocks. This change may represent additional servers on the Internet. As described above, we know the always/sometimes-stable border is sensitive to observation duration, so future work is required to understand whether these shifts are meaningful.

6. CONCLUSION

We have shown that active probes can identify how Internet addresses are used, confirming that contiguous addresses are often used similarly. We have validated our claims at USC, against randomly selected Internet blocks, and over multiple years. Within the constraints of active probing, our approach provides a new tool to understand Internet use and trends.

Acknowledgments

We would like to thank Brian Yamaguchi and Maureen Dougherty for the valuable USC campus network administrative information. We thank Roque Gagliano for discussions about LACNIC, and Anja Feldman about Deutsche Telekom. We thank Yuri Pradkin for the probing software and ping data collection used here, and Paul Vixie and ISC for providing their DNS data. We thank the anonymous reviewers, John Wroclawski, and our shepherd, Yinglian Xie, for suggestions that improved the paper's writing.

This material is based on work partially supported by the U.S. Department of Homeland Security contract number NBCHC080035 (”LANDER-2007”), and by the National Science Foundation under grant number CNS-0626696, "NeTS-NBD-SGER: Map/Reduce for Network Traffic Analysis". The conclusions of this work are those of the authors and do not necessarily reflect the views of DHS or NSF.

7. REFERENCES

[1] American Registry for Internet Numbers. RIR statistics exchange format. Technical report, ARIN, September 2008. (Retrieved January, 2009).

[2] APNIC. IPv4 initial delegation criteria. web page `http://www.apnic.net/services/apply-for-resources/check-your-eligibility`, May 2010.

[3] ARIN. Template: ARIN-NET-ISP-4.1. web page `https://www.arin.net/resources/templates/net-isp.txt`, April 2010.

[4] Xue Cai and John Heidemann. Understanding block-level address usage in the visible Internet. In *Proc. of ACM SIGCOMM*, page to appear, New Delhi, India, August 2010. ACM.

[5] Xue Cai and John Heidemann. Understanding Block-level Address Usage in the Visible Internet (Extended). Technical Report ISI-TR-2009-665, USC/ISI, July 2010.

[6] Xenofontas Dimitropoulos, Dmitri Krioukov, Marina Fomenkov, Bradley Huffaker, Young Hyun, kc claffy, and George Riley. AS Relationships: Inference and Validation. *ACM CCR*, 37(1):29–40, January 2007.

[7] R. Droms. Dynamic Host Configuration Protocol. RFC 2131, March 1997.

[8] Brian Eriksson, Paul Barford, and Robert Nowak. Network Discovery from Passive Measurements. In *ACM SIGCOMM*, pages 291–302, Seattle, Washigton, USA, August 2008. ACM.

[9] Kevin Fall. A Delay-Tolerant Network Architecture for Challenged Internets. In *Proc. of ACM SIGCOMM*, pages 27–34, Karlsruhe, Germany, August 2003. ACM.

[10] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On Power-Law Relationships of the Internet Topology. In *Proc. of ACM SIGCOMM*, pages 251–262, Cambridge, MA, USA, September 1999. ACM.

[11] Sally Floyd and Vern Paxson. Difficulties in simulating the Internet. *ACM/IEEE Trans. on Networking*, 9(4):392–403, August 2001.

[12] Michael Freedman, Mythili Vutukuru, Nick Feamster, and Hari Balakrishnan. Geographic Locality of IP Prefixes. In *Proc. of ACM IMC*, Berkeley, CA, October 2005. ACM.

[13] V. Fuller, T. Li, J. Yu, and K. Varadhan. Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy. RFC 1519, September 1993.

[14] Lixin Gao. On Inferring Automonous System Relationships in the Internet. *ACM/IEEE Trans. on Networking*, 9(6):733–745, December 2001.

[15] John Heidemann, Yuri Pradkin, Ramesh Govindan, Christos Papadopoulos, Genevieve Bartlett, and Joseph Bannister. Census and Survey of the Visible Internet. In *Proc. of ACM IMC*, pages 169–182, Vouliagmeni, Greece, October 2008. ACM.

[16] Geoff Huston. IPv4 Reports. `http://bgp.potaroo.net/index-ale.html`, April 2009.

[17] Internet Software Consortium. Internet Domain Survey. web page `http://www.isc.org/solutions/survey`, January 2007.

[18] Van Jacobson. pathchar—a tool to infer characteristics of Internet paths. MSRI talk, April 1997.

[19] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, Englewood Cliffs, NJ, 1988.

[20] kc claffy, Young Hyun, Ken Keys, Marina Fomenkov, and Dmitri Krioukov. Internet Mapping: from Art to Science. In *IEEE DHS CATCH*, page to appear, Washington, US, March 2009. IEEE.

[21] Srinivasan Keshav. A Control-Theoretic Approach to Flow Control. In *Proc. of ACM SIGCOMM*, pages 3–16, Zurich, Switzerland, September 1991. ACM.

[22] Manas Khadilkar, Nick Feamster, Matt Sanders, and Russ Clark. Usage-based DHCP Lease Time Optimization. In *Proc. of 7th ACM IMC*, pages 71–76, San Diego, California, USA, October 2007. ACM.

[23] LACNIC. LACNIC Policy Manual (v1.2 - 11/03/2009). `http://www.lacnic.net/en/politicas/manual3.html`, April 2009.

[24] Gordon Lyon. nmap. computer software at `http://insecure.org/nmap/`, September 1997.

[25] Xiaoqiao Meng, Zhiguo Xu, Beichuan Zhang, Geoff Huston, Songwu Lu, and Lixia Zhang. IPv4

Address Allocation and the BGP Routing Table Evolution. *ACM CCR*, 35(1):71–80, January 2005.

[26] P. Mockapetris. Domain names—concepts and facilities. RFC 1034, November 1987.

[27] Wolfgang Mühlbauer, Olaf Maennel, Steve Uhlig, Anja Feldmann, and Matthew Roughan. Building an AS-Topology Model that Captures Route Diversity. In *Proc. of ACM SIGCOMM*, pages 195–204, Pisa, Italy, September 2006. ACM.

[28] Jon Postel. Internet Protocol. RFC 791, September 1981.

[29] Regional Internet Registry. Resource ranges and geographical data. web page `ftp://ftp.afrinic.net/pub/stats/afrinic/`, `ftp://ftp.apnic.net/pub/stats/apnic/`, `ftp://ftp.arin.net/pub/stats/arin/`, `ftp://ftp.lacnic.net/pub/stats/lacnic/`, `ftp://ftp.ripe.net/ripe/stats/`, June 2007.

[30] J. Rosenberg, J. Weinberger, C. Huitema, and R. Mahy. STUN—Simple Traversal of User Datagram Protocol Through Network Address Translators. RFC 3489, December 2003.

[31] Rob Sherwood, Adam Bender, and Neil Spring. DisCarte: A Disjunctive Internet Cartographer. In *Proc. of ACM SIGCOMM*, pages 303–314, Seattle, Washigton, USA, August 2008. ACM.

[32] Lakshminarayanan Subramanian, Sharad Agarwal, Jennifer Rexford, and Randy H. Katz. Characterizing the Internet Hierarchy From Multiple Vantage Points. In *Proc. of IEEE Infocom*, pages 618–627, New York, NY, USA, June 2002. IEEE.

[33] Matthew Sullivan and Luis Munoz. Suggested generic DNS naming schemes for large networks and unassigned hosts. Work in progress (Internet draft `draft-msullivan-dnsop-generic-naming-schemes-00.txt`, April 2006.

[34] Ionut Trestian, Supranamaya Ranjan, Aleksandar Kuzmanovic, and Antonio Nucci. Unconstrained Endpoint Profiling (Googling the Internet). In *Proc. of ACM SIGCOMM*, pages 279–290, Seattle, Washigton, USA, August 2008. ACM.

[35] USC/LANDER project. Internet Addresses Survey dataset, PREDICT ID `USC-LANDER/internet_address_survey_reprobing_it16w-20070216`. web page `http://www.isi.edu/ant/lander`, February 2007.

[36] USC/LANDER project. Internet Addresses Survey dataset, PREDICT ID `USC-LANDER/internet_address_survey_reprobing_it17w-20070601`. web page `http://www.isi.edu/ant/lander`, June 2007.

[37] USC/LANDER project. Internet Addresses Survey dataset, PREDICT ID `USC-LANDER/survey_validation_usc-20070813`. web page `http://www.isi.edu/ant/lander`, August 2007.

[38] USC/LANDER project. Internet Addresses Survey dataset, PREDICT ID `USC-LANDER/internet_address_survey_reprobing_it30w-20091223`. web page `http://www.isi.edu/ant/lander`, December 2009.

[39] USC/LANDER project. Internet Addresses Survey dataset, PREDICT ID `USC-LANDER/internet_address_survey_reprobing_it31w-20100208`. web page `http://www.isi.edu/ant/lander`, February 2010.

[40] Yinglian Xie, Fang Yu, Kannan Achan, Eliot Gillum, Moises Goldszmidt, and Ted Wobber. How Dynamic are IP Addresses? In *Proc. of ACM SIGCOMM*, page to appear, Kyoto, Japan, August 2007. ACM.

APPENDIX

A. DETAILS ABOUT SURVEYS AT DIFFERENT DATES

Due to space limitations, the conference paper [4] provided details for 10-day analysis of *IT17ws* in Table 3. This appendix provides data for the 6-day subset of *IT17ws* (Table 11) to compare to the 6 days of *IT16ws* (Table 12) and 6 days of *IT31ws* (Table 13). These results were discussed in Section 5.3 to establish that the results are generally consistent across adjacent surveys of the same length, and to describe the evolution over surveys years apart.

We also provide all 14 days of *IT31ws* in Table 14 for comparison to the 6-day subset.

B. EXAMINING THE (A,V,U*) SPACE

Section 3.4 defined our ping-observable categories based on the (A, V, U^*) values of blocks. To develop an understanding of how these metrics help categorize the Internet, Figure 7 shows the density plot of (A, V, U^*) space separated in three planes. For each plot, we create 100 bins for each of two parameters, then count the number of /24 blocks identified in *IT17ws* that fall into that bin with any value of the third parameter.

All of the planes show blocks with many different values, providing no definitive clusters. However, there are concentrations in some areas of some planes, even though there are a few blocks in between those concentrations. The (A, V) plane shows two concentrations, with a portion of blocks tend to gather around $(A, V, U^*) = (0.975, 0.005, *)$, showing highly available and highly stable behavior. We classify most of them into always-stable blocks. Another portion of blocks tend to gather around $(A, V, U^*) = (0.050, 0.005, *)$ which exhibit highly underutilized behavior. We classify them into underutilized blocks. The rest blocks are distributed

size			sometimes-					blocks	addresses
pfx	addrs	always-stable	stable	intermittent	underutilized	classifiable (100%)	unclassifiable	*[100%]*	
/24	256	2,001 (23%)	2,328 (27%)	2,399 (28%)	1,832 (21%)	8,560*	3,522 *[29%]*	12,082	3,092,992
/25	128	376 (%)	553 (%)	292 (%)	222 (%)	1,443*	965 *[40%]*	2,408	
/26	64	401 (%)	602 (%)	371 (%)	241 (%)	1,615*	824 *[34%]*	2,439	
/27	32	550 (%)	834 (%)	485 (%)	319 (%)	2,188†	864 *[28%]*	3,052	
/28	16	824 (%)	1,258 (%)	913 (%)	616 (%)	3,611†	1,219 *[25%]*	4,830	
/29	8	2,506 (%)	3,033 (%)	2,279 (%)	2,129 (%)	9,947†	0	9,947	
/30	4	4,052 (%)	5,492 (%)	4,881 (%)	3,657 (%)	18,082†	0	18,082	
/31	2	5,933 (%)	10,431 (%)	8,757 (%)	4,844 (%)	29,965†	0	29,965	
/32	1	53,433 (30%)	40,266 (22%)	47,497 (26%)	38,254 (21%)	179,450†	0	179,450	
entire *IT17ws* dataset:		(1,602,412 addrs. in non-responsive blocks) + (4,123,540 in responsive blocks)						22,367	5,725,952

Table 11: Number of blocks of each size in *IT17ws* (6 days). Unclassifiable percentages relative to all blocks; other percentages relative to classifiable blocks. Asterisks: consistent blocks, daggers: non-consistent.

size			sometimes-					blocks	addresses
pfx	addrs	always-stable	stable	intermittent	underutilized	classifiable (100%)	unclassifiable	*[100%]*	
/24	256	2,173 (26%)	2,007 (24%)	2,409 (28%)	1,915 (23%)	8,504*	3,518 *[29%]*	12,022	3,077,632
/25	128	424 (%)	489 (%)	339 (%)	215 (%)	1,467*	965 *[40%]*	2,432	
/26	64	421 (%)	610 (%)	409 (%)	218 (%)	1,658*	770 *[32%]*	2,428	
/27	32	547 (%)	808 (%)	532 (%)	326 (%)	2,213†	805 *[27%]*	3,018	
/28	16	899 (%)	1,246 (%)	962 (%)	634 (%)	3,741†	1,169 *[24%]*	4,910	
/29	8	2,644 (%)	2,909 (%)	2,441 (%)	2,075 (%)	10,069†	0	10,069	
/30	4	4,513 (%)	5,514 (%)	5,031 (%)	3,836 (%)	18,894†	0	18,894	
/31	2	6,278 (%)	10,327 (%)	8,581 (%)	5,092 (%)	30,278†	0	30,278	
/32	1	54,916 (31%)	39,830 (22%)	45,879 (26%)	39,065 (22%)	179,690†	0	179,690	
entire *IT16ws* dataset:		(1,609,610 addrs. in non-responsive blocks) + (4,115,830 in responsive blocks)						22,365	5,725,440

Table 12: Number of blocks of each size in *IT16ws* (6 days). Unclassifiable percentages relative to all blocks; other percentages relative to classifiable blocks. Asterisks: consistent blocks, daggers: non-consistent.

size			sometimes-					blocks	addresses
pfx	addrs	always-stable	stable	intermittent	underutilized	classifiable (100%)	unclassifiable	*[100%]*	
/24	256	2,459 (33%)	1,363 (18%)	2,023 (27%)	1,605 (22%)	7,450*	4,045 *[35%]*	11,495	2,942,720
/25	128	449 (%)	257 (%)	179 (%)	112 (%)	997*	1007 *[50%]*	2,004	
/26	64	540 (%)	373 (%)	234 (%)	119 (%)	1,266*	813 *[39%]*	2,079	
/27	32	741 (%)	606 (%)	370 (%)	201 (%)	1,918†	883 *[32%]*	2,801	
/28	16	1244 (%)	919 (%)	688 (%)	451 (%)	3,302†	1,339 *[29%]*	4,641	
/29	8	3,494 (%)	2,143 (%)	1,776 (%)	1,431 (%)	8,844†	0	8,844	
/30	4	4,640 (%)	4,209 (%)	3,774 (%)	2,771 (%)	15,394†	0	15,394	
/31	2	5,847 (%)	9,104 (%)	7,335 (%)	3,783 (%)	26,069†	0	26,069	
/32	1	79,283 (40%)	40,136 (20%)	44,116 (22%)	34,785 (18%)	198,320†	0	198,320	
entire *IT31ws* dataset:		(1,849,294 addrs. in non-responsive blocks) + (3,878,962 in responsive blocks)						22,376	5,728,256

Table 13: Number of blocks of each size in *IT31ws* (6 days). Unclassifiable percentages relative to all blocks; other percentages relative to classifiable blocks. Asterisks: consistent blocks, daggers: non-consistent.

size			sometimes-			classifiable	unclassifiable	blocks	addresses
pfx	addrs	always-stable	stable	intermittent	underutilized	(100%)		*[100%]*	
/24	256	1,735 (23%)	1,676 (22%)	2,300 (30%)	1,994 (26%)	7,705*	3,606 *[32%]*	11,311	2,895,616
/25	128	313 (%)	340 (%)	174 (%)	145 (%)	972*	942 *[49%]*	1,914	
/26	64	361 (%)	487 (%)	231 (%)	181 (%)	1,260*	761 *[38%]*	2,021	
/27	32	514 (%)	686 (%)	396 (%)	289 (%)	1,885†	869 *[32%]*	2,754	
/28	16	878 (%)	1,236 (%)	784 (%)	587 (%)	3,485†	1,286 *[27%]*	4,771	
/29	8	2,016 (%)	3,188 (%)	1,897 (%)	1,906 (%)	9,007†	0	9,007	
/30	4	3,020 (%)	6,231 (%)	4,007 (%)	3,377 (%)	16,635†	0	16,635	
/31	2	1,113 (%)	12,806 (%)	7,229 (%)	4,269 (%)	25,417†	0	25,417	
/32	1	81,320 (38%)	49,609 (23%)	42,767 (20%)	38,969 (18%)	212,665†	0	212,665	
entire *IT31ws* dataset:		(1,891,745 addrs. in non-responsive blocks) + (3,836,511 in responsive blocks)						22,376	5,728,256

Table 14: Number of blocks of each size in *IT31ws* (14 days). Unclassifiable percentages relative to all blocks; other percentages relative to classifiable blocks. Asterisks: consistent blocks, daggers: non-consistent.

size			sometimes-			classifiable	unclassifiable	blocks	addresses
pfx	addrs	always-stable	stable	intermittent	underutilized	(100%)		*[100%]*	
/24	256	1,884 (25%)	1,522 (20%)	2,256 (29%)	2,008 (26%)	7,670*	3,451 *[31%]*	11,121	2,846,976
/25	128	358 (%)	291 (%)	161 (%)	166 (%)	976*	846 *[46%]*	1,822	
/26	64	435 (%)	358 (%)	217 (%)	211 (%)	1,221*	594 *[33%]*	1,815	
/27	32	547 (%)	568 (%)	346 (%)	299 (%)	1,760†	729 *[29%]*	2,489	
/28	16	856 (%)	988 (%)	730 (%)	648 (%)	3,222†	1,091 *[25%]*	4,313	
/29	8	1,838 (%)	2,306 (%)	1,834 (%)	2,127 (%)	8,105†	0	8,105	
/30	4	2,731 (%)	4,624 (%)	3,978 (%)	3,478 (%)	14,811†	0	14,811	
/31	2	2,890 (%)	8,915 (%)	7,110 (%)	4,377 (%)	23,292†	0	23,292	
/32	1	98,867 (44%)	45,016 (20%)	41,529 (18%)	40,368 (18%)	225,780†	0	225,780	
entire *IT30ws* dataset:		(1,988,080 addrs. in non-responsive blocks) + (3,741,456 in responsive blocks)						22,381	5,729,536

Table 15: Number of blocks of each size in *IT30ws* (14 days). Unclassifiable percentages relative to all blocks; other percentages relative to classifiable blocks. Asterisks: consistent blocks, daggers: non-consistent.

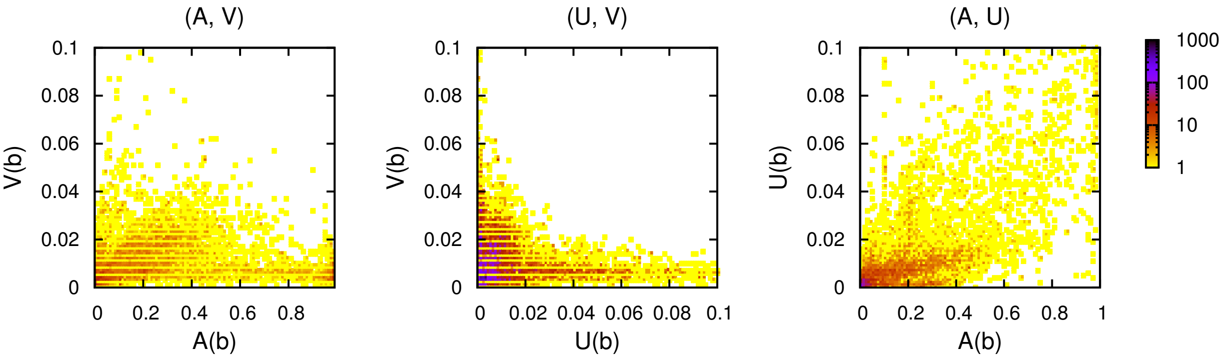

Figure 7: Density plots of /24 blocks in *IT17ws* across each of the A/V, U/V, A/U planes.

between $(A, V, U^*) = (0.100 - 0.400, 0.075 - 0.022, *)$, with no obvious boundary to differentiate sometimes-stable and intermittent blocks on (A, V) plane. Instead, we inspect the (A, M) and (M, V) planes to split these apart. Even there, we do not see a sharp boundary. However, we place a line at $U = 0.026$ ($U^* = 6$ hours) to classify sometimes-stable ($U^* \geq 6$ hours) and intermittent ($U^* \leq 6$ hours) blocks.

C. TRAINING AND HOSTNAME-INFERRED USAGE CATEGORIZATION

Our methodology takes data about use of public addresses and produces five ping-observable categories. We would like to relate those categories to terms that are more meaningful to network operators, and to find what root causes correspond to and potentially cause blocks to be intermittent or underutilized.

Determining the operational characteristics of a network is quite challenging, however. In some cases we are able to discuss network policy with the operations staff to confirm our assumptions; we will use this approach to validate our conclusions against a large campus network in Section 5. However, such observations may be biased by the policies of a single institution. We would like to also draw data from the Internet at large, but it is infeasible to contact operations for large parts of the network. While tools such as nmap [24] can extract significant information from a network through sophisticated active probing, their use is easy confused with hostile network activity by many network operations.[3]

Hostnames are a source of data that provides some information about how public computers are used—many hostnames contain keywords such as "www", "dynamic", or "dsl". Wide hostnames collection is also feasible: many Internet hosts suggest reverse DNS lookup [26], reverse lookup occurs commonly as part of normal operation and so is unlikely to be seen as hostile. The Internet Systems Consortium has collected full tables of reverse DNS regularly since 1994 [17] and makes it available for a nominal fee.

We next describe how we map hostnames to 15 *hostname-inferred usage categories*, and how this data corresponds to our five ping-observable categories.

While one might study the Internet using hostname data alone, without ping data, we believe the information complements each other. About half addresses that are used lack reverse hostnames, and about 49% of hostnames lack meaningful keywords, and reverse names may not represent the computer's true use (for hosts with multiple names, where often the reverse name is automatically assigned) so we think hostnames alone are not sufficient.

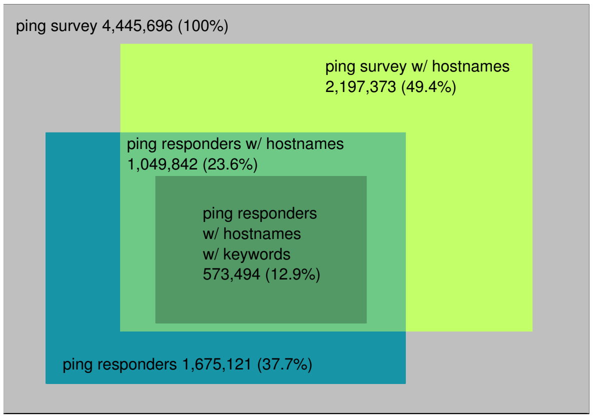

Figure 8: Our Investigation Targets: IP addresses ever responded in *IT17wrs* and have meaningful hostnames (with keywords). It is the middle part with 573,494 addresses in this figure.

C.1 Hostname-inferred Usage Categories

Although hostnames are not perfect, we believe they provide a useful dataset to compare against our ping-observable categories. We use ISC survey 17 [17], taken slightly before our primary ping survey [36].

Figure 8 shows the overlap of these datasets. shows our investigation targets. We begin with the 4.4M IP addresses probed the ping survey Nearly half of these (2.2M) have hostnames in the ISC reverse DNS survey. Of the 1.6M ping addresses that respond, we consider the 1.0M addresses that also have hostnames. We then focus on the 573,494 of those that have identifiable keywords in their hostname (12.9% of all addresses in the ping survey).

We follow recommendations that were proposed as standard naming conventions for Internet hosts [33] and that occur in 2000 or more hosts in our dataset. Although these were neither approved by the IETF, nor would the be mandatory even if approved, these terms do appear in about one-quarter of reverse hostnames. From their recommendations we define 15 *hostname-inferred usage categories* as shown in Table 16.

Figure 9 shows the count of hostnames in each category. The sum exceeds 573k because these categories partially overlap, and a single hostname may be in multiple categories. For example, some providers label DSL addresses with both DSL and static or dynamic. We see that access links keywords (DSL, dial, etc.) are very common, occurring in 51% of hostnames, and allocation types (static, dynamic, etc.) occur in about 22% of hostnames in *ping survey w/ hostnames*.

To provide some understanding the number of hostnames with multiple keywords, we subdivide each cate-

[3] Widescale nmap use would place us in contact with additional operations staff, but perhaps not on ideal terms.

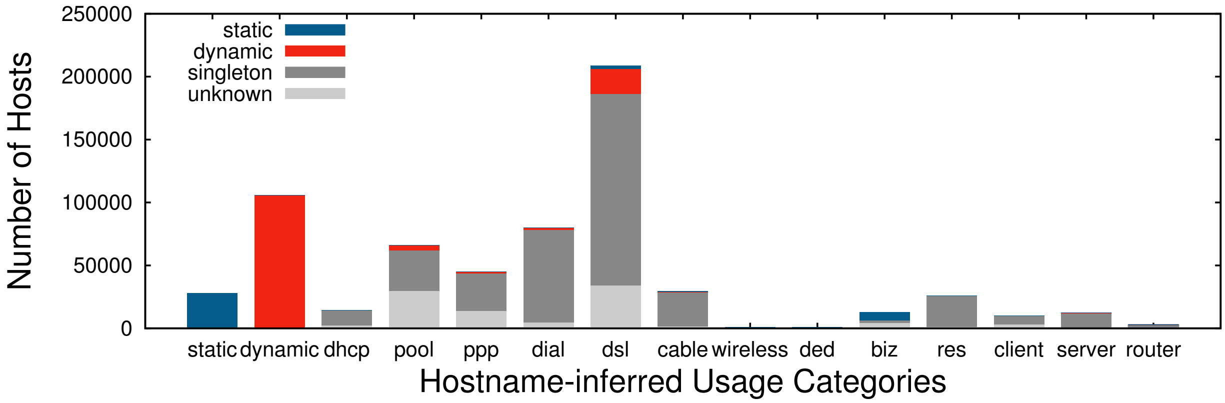

Figure 9: Numbers of hostname-inferred usage categories, with colors indicating those that also have allocation types.

group	category	keywords	count
allocation	static	static	28,137
	dynamic	dynamic, dyn	105,882
	dhcp	dhcp	14,290
	pool	pool, pond	66,009
	ppp	ppp	44,729
access link	dial	dial, modem	80,090
	dsl	dsl	208,682
	cable	cable	29,761
	wireless	wireless, wifi	910
	ded	ded, dedicated	733
consumer	biz	business, biz	12,999
	res	res, resident	25,847
	client	client	9,994
server	server	server, srv, svr, mx, mail, smtp, www, ns, ftp	12,568
	router	router, rtr, rt, gateway, gw	2,850

Table 16: Categories of hostname-derived usage.

gory by those that also contain static, dynamic, or any other additional keyword. Several groups types often have an additional indication of allocation type: while 10% of dsl are labeled dynamic (1.2% static), 50% of biz are labeled static. These secondary attributes reveal some technology trends: the ratio of dial also with static or dynamic types is around 1:17, while for DSL it is 1:8 suggesting increased use of static addresses in always-on DSL lines. For cable the ratio is 1:1, but the fraction of cables with an additional type is small enough that drawing conclusions may be risky.

C.2 Relating Hostname-Inferred to Ping-Observable Categories

Our goal in evaluating hostnames is to use them to understand and train our ping-observable categories. We next compare the two to see when of our observations (A, V, U^*) are correlated with hostname-inferred usage categories.

Figures 10 shows cumulative distribution functions of each observation against each hostname-inferred type. This data will prove essential to understand the root network causes of address underutilization and locations of dynamic addresses; we therefore defer a detailed discussion of this data to Sections [and 4.2][4.3].

Taken together, though, these graphs support the third assumption of our paper, that **patterns of probe responses can suggest address usage**. This assertion is supported because hostname-inferred categories (our approximation of usage) show fairly distinct distributions, particularly in availability and median-up (Figure 10). As a specific example, The left graph of Figure 10 shows that availability of more than 50% dial addresses is smaller than 0.1, while the A of more than 80% server addresses is larger than 0.95.

While the bulk of dial and server addresses are quite different, a few dial addresses have with reasonably large A (5.4% have $A > 0.5$), and a moderate number of servers have poor availability (about 10% have $A < 0.15$). We conclude that, while ping-observable metrics are reasonable predictors of usage, they are not exact, and any estimates will have fairly large error bounds. Perhaps this result is consistent with previous observations about the great variability of the Internet [11].

Finally, the observations in these CDFs help define our thresholds for ping-observable classes (Section 3.4). The sharp knee at $A = 0.1$ in the left graph Figure 10 suggests $\alpha_L = 0.1$. Based on V in the middle graph, we select $\beta = 0.0016$ to separate most servers and stable uses from less stable. The sharp knee at around $U^* = 6$ hours in the right graph suggests this value for γ, This cutoff helps separate addresses which are not always-stable and not underutilized to two categories: sometimes-stable and intermittent.

We observe one anomaly in U^*: the step in the right graph of Figure 10 around hour 120 is an artifact caused by hosts that have high A (nearly 1.0), but a brief outage. This outage gives them $V^* = 2$, and U^* of half our probing duration.

Based on these thresholds, Figure 11 and Table 17

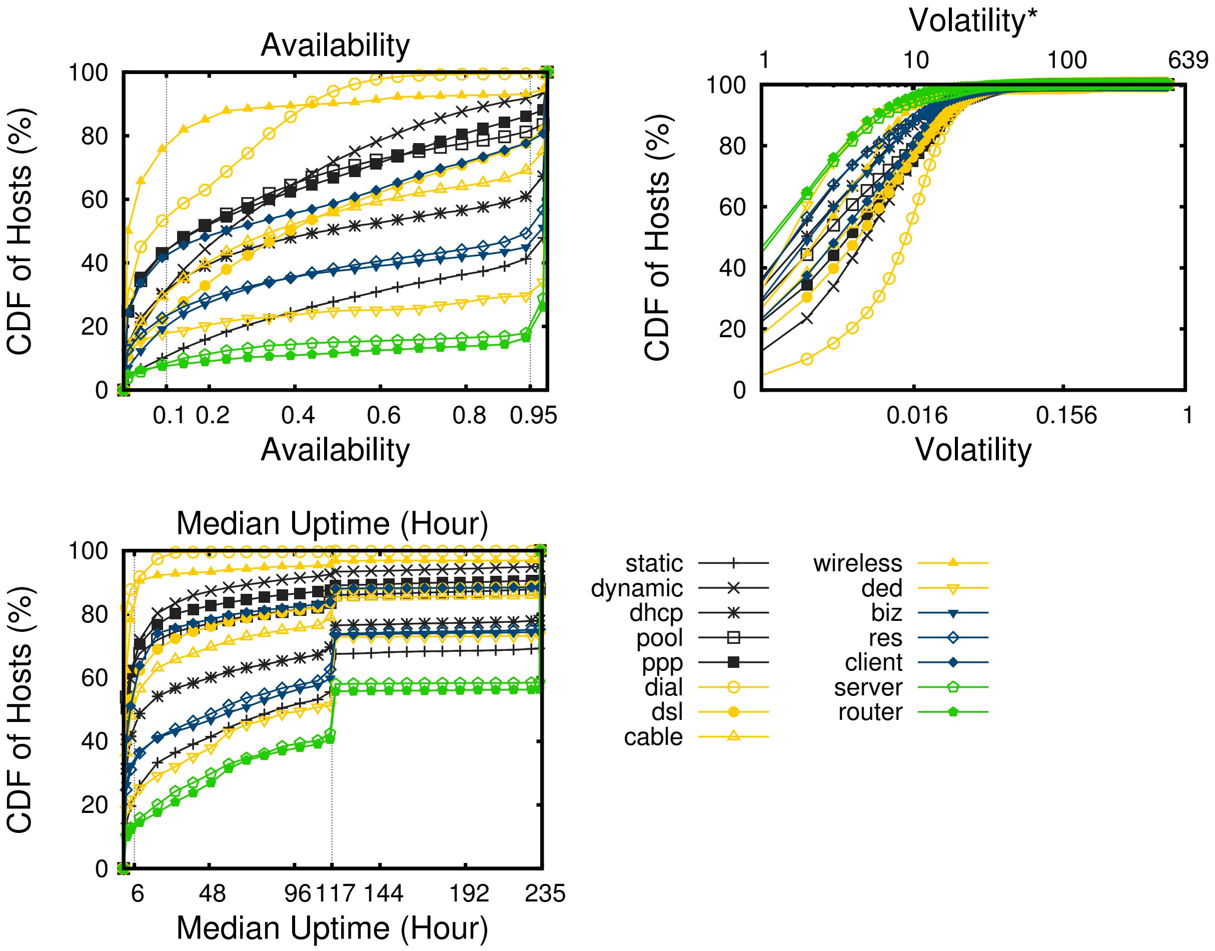

Figure 10: CDF of address availability (A), volatility (V) and median-up duration (U^*) by hostname-inferred categories in *IT17ws*.

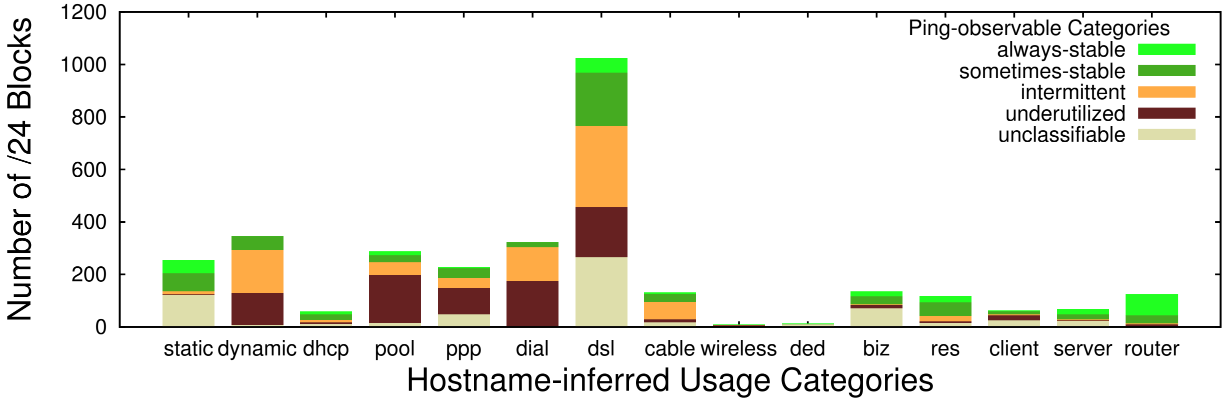

Figure 11: Relationship of ping-observed categories to hostname-inferred categories in *IT17ws*.

ping-observable category	**hostname-inferred usage category**
always-stable	router, server, (static), ded, (biz), (dhcp), (res)
sometimes-stable	res, static, biz, dhcp, (server), ded, (client), (cable), (dsl), (router), (ppp)
intermittent	cable, dynamic, dsl, (wireless), (dial), (ppp), (dhcp)
underutilized	pool, wireless, ppp, dial, client, (dynamic), (ded), (dsl)

Table 17: The mapping from the 15 hostname-inferred usage categories to 4 ping-observable categories. hostname-inferred usage category without (parentheses) is dominate.

map the 15 hostname-inferred usage categories to the ping-observable categories (Section 3.4).