Detecting Internet Outages with Precise Active Probing (extended)

USC/ISI Technical Report ISI-TR-678b

February 2012, updated May 2012 *

Lin Quan John Heidemann Yuri Pradkin
USC/Information Sciences Institute
{linquan, johnh, yuri}@isi.edu

ABSTRACT

Parts of the Internet are down every day, from the intentional shutdown of the Egyptian Internet in Jan. 2011 and natural disasters such as the Mar. 2011 Japanese earthquake, to the thousands of small outages caused by localized accidents, and human error, maintenance, or choices. Understanding these events requires efficient and accurate detection methods, motivating our new system to detect network outages by active probing. We show that a *single computer* can track outages across the entire analyzable IPv4 Internet, probing a sample of 20 addresses in all 2.5M responsive /24 address blocks. We show that our approach is *significantly more accurate* than the best current methods, with 31% fewer false conclusions, while providing 14% greater coverage and requiring about the same probing traffic. We develop new algorithms to identify outages and cluster them to events, providing the first visualization of outages. We carefully validate our approach, showing consistent results over two years and from three different sites. Using public BGP archives and news sources we confirm 83% of large events. For a random sample of 50 observed events, we find 38% in partial control-plane information, reaffirming prior work that small outages are often not caused by BGP. Through controlled emulation we show that our approach detects 100% of full-block outages that last at least twice our probing interval. Finally, we report on Internet stability *as a whole*, and the size and duration of typical outages, using core-to-edge observations with much larger coverage than prior mesh-based studies. We find that about 0.3% of the Internet is likely to be unreachable at any time, suggesting the Internet provides only 2.5 "nines" of availability.

1. INTRODUCTION

End-to-end reachability is a fundamental service of the Internet. Network outages break protocols based on point-to-point communication and often harm the user's experience of Internet applications. Replication and content delivery networks strive to cover up outages, but in spite of decades of research on network reliability, Internet outages are still pervasive, ranging from minutes to hours and days. Outages are triggered by system, link or router breakdowns [32, 42]. Causes of these failures include natural disasters [32, 39], human error [31], and political upheavals [9–11, 41]. On occasion, routing changes can cause user-visible problems as the network reconfigures [27, 28, 40].

The contributions of our work are to provide a new method that can systematically find *outages*, unreachable blocks of adjacent network addresses, for *all of the analyzable IPv4 Internet*—a method that provides better accuracy and coverage than existing approaches, particularly for small events. Second, we carefully validate our approach comparing onset and duration of outages to root causes both for widely publicized events such as the Jan. 2011 Egypt outage, and for randomly sampled small outages. And finally, to provide a statistical characterization of Internet outages as a whole and for specific blocks, extending prior evaluation using meshes to cover the entire network edge.

Our first contribution is our new approach to active probing, showing that a *single computer* can track outages over the entire, analyzable IPv4 Internet (the 2.5M /24 blocks that are suitable for our analysis, see §4.4). Like prior work [20], we send active ICMP probes to addresses of each /24 block every 11 minutes. Unlike it, we develop a new approach of *precise* probing that carefully selects a subset of blocks and addresses per block to reduce probing traffic by a factor of 75, while retaining more than 90% accuracy for outage detection (§4). We develop a new method to distill this data into block-level outage reports (§3.2), defining an outage as a sharp change in block responsiveness relative to recent behavior. We interpret the observations of our system

*This technical report updates ISI-TR-678 with substantial editorial changes and new experiments in Section 5.6.2.

The authors are partially supported by the US DHS, contract number NBCHC080035, and Heidemann by the NSF, grant number CNS-0626696. The conclusions of this work are those of the authors and do not necessarily reflect the views of DHS or NSF.

by correlating block-level outages to discover network-wide events with two new clustering algorithms. The first groups outages in two dimensions, time and space, to provide a general understanding of network behavior, associate outages to countries, and provide *the first visualization* of outages (§3.3). The second, more general algorithm, finds network-wide events from the start- and end-times of block-level outages (§3.4).

Several prior systems study network outages—the new contribution of our approach is *significantly greater accuracy* than the best current active methods, and operation from a single computer with about the same probing traffic. Unlike control-plane studies [27, 33], we detect outages that are not seen in the routing system (§5.3), expanding the result observed by Bush et al. [5]. Unlike previous data-plane studies using active probing, including DIMES [37], iPlane [30], Hubble [23], and SCORE [25, 26], our block-level measurements are considerably more accurate at detecting core-to-edge outages. Comparisons to Hubble show that *our approach reduces the number of false conclusions* by 31% compared to approaches probing a single representative per /24 block, with about the same traffic (§5.6). Unlike network tomography that focused localizing outages [12, 15, 22, 25, 26], we instead focus on tracking core-to-edge reachability; our work could serve as a trigger for such localization methods. We remove outages near our vantage points to correct for correlated error (§6.3). Of course, our approach shares the limitation of all those based on active probing: it can only report on the visible Internet, those willing-to-respond blocks; currently we can monitor about 2.5M /24 blocks, about one-seventh more coverage than Hubble. Recent work has combined backscatter with routing information to characterize large outages [13, 14]; we show that active probing complements this work and is critical to detect small outages and provide Internet-wide statistics. We cover related work more generally in §2.

The second contribution of our work is to validate the accuracy of active probing for outage detection (§5). Even though we probe from a single location, we draw on data sources taken from three vantage points in California, Colorado, and Japan, to show our results are largely insensitive to location (§5.5). We study more than 30 observations taken over more than two years, using 2 week surveys of all addresses in a 1% sample of Internet blocks [20], and a 24-hour measurement taken across all suitable /24 blocks in the Internet in Sep. 2011 to show that our results are stable over time. We validate our approach with BGP archives and news sources, for selected large events and a random sample of 50 observed events. We confirm 5 of 6 large events (83%, §5.2), including the Jan. 2011 Egyptian outage, the Mar. 2011 Japanese earthquake, and equally large but less newsworthy events. Our random sample of all events confirm prior work by Bush et al. [5] showing that small outages often do not appear in control-plane messages, since partial control-plane information shows only 38% of small outages we observe (§5.3). We emulate outages of controlled length to investigate false availabilities. We miss very short outages, and detect 100% of full-block outages that last at least twice our probing interval (§5.4).

Our final contribution is to evaluate Internet stability as a whole (§6). We show that, on average, about 0.3% of the Internet is inaccessible at any given time. The Internet blocks have around 99.7–99.8% availability, only about 2.5 "nines", as compared to the "five nines" telephone industry. While prior work has studied paths between meshes of hundreds of computers and thousands of links, and anecdotes about the Internet as a whole abound, we provide much broader coverage with quantitative data about *all* responsive 2.5M edge /24 blocks. We believe these statistics can establish a baseline of Internet reliability, allowing future comparisons of Internet reliability across ISPs or geography.

Data from this paper is available at no charge [43].

2. RELATED WORK

We next review prior studies of network stability based on control-plane, data-plane, and other observations.

2.1 Control-plane Studies

Several prior efforts use control-plane data to study Internet outages. Markopoulou et al. use IS-IS update messages to classify failures in Sprint's network. They report percentages of outages categorized by layer (maintenance, router, and optical) [33]. Like them, we use control-plane data (BGP archives) and out-of-band information (news reports logs), but we use it only to validate our results; our outage discovery uses data-plane probes exclusively.

Omni employs a server in each Autonomous System (AS) that maintains an AS-level forwarding table to diagnose routing changes, and overcoming the limitations of public routing information [38]. Omni deployment therefore requires wide adoption to get good coverage. Our work uses centrally-collected measurement and analysis, easing deployment.

Labovitz et al. induce controlled routing failures and study their impact on end-to-end network performance [27]. We find outages directly, including many not due to routing and invisible to the control plane.

BGP misconfiguration can also be a cause of outages. Mahajan et al. study routing messages and contact network operators about BGP misconfiguration [31]. They also use active probing to determine the impact of misconfiguration on connectivity. They report that 0.2% to 1% of prefixes suffer from misconfiguration each day. We confirm their results on Internet reachability, find-

ing about 0.3% of the Internet blocks are out at any instant. Our methodology allows detection of all types of outages (not just BGP-triggered ones), and finds outages not visible to the control plane [5, 23].

Control-plane studies of reachability are necessarily indirect and thus inherently limited, as discussed by Bush et al. [5]. There exist ways to alleviate such limits, for example, Huang et al. [21] use a multivariate method with many BGP sources to detect network disruptions. However, we show that data-plane measurements are necessary to detect non-control outages, and so use control-plane information only for validation.

2.2 Data-plane Studies

Direct data-plane measurements can be more accurate than those of control-plane. Choffnes et al. collect information from end systems to detect service-level network events [8]. Our work is different in that we probe to the network edge and don't require extra software to run on the end systems. Our system is independent of operating systems in the sense that information from all kinds of end systems can be utilized, as long as they respond to pings.

Very close to our work, the Hubble system uses continuous probes to the .1 of each routed /24 block, to find potential Internet outages [23]. We instead probe multiple or all addresses in each /24 block. We study the tradeoff between sampling and accuracy (§5.6.1) and show our use of multiple representatives per block greatly reduces the number of false conclusions about outages (§5.6.2). We also describe new algorithms for clustering outages for visualization and into network-wide events.

Cunha et al. run multiple probes to confirm a link failure and location. They analyze the benefits of numbers of probes, and improve accuracy with minimal probing overhead [12]. We also study the tradeoff in probe volume against accuracy (§5.6.1), but focus on end-system outage detection rather than specific link failures.

Bush et al. study the reachability of Internet address space using traceroute to detect incorrect filtering [4], and find biases in reachability experiments [5]. We provide additional evidence supporting their observation that default routes are widely used and that control-plane measurements underestimate outages.

2.3 Client-supported Analysis

Unlike the above, centrally-run methods, one can also use client-side measurement support. Several prior groups have used meshes of measurement computers [1, 17, 26, 35]. Such experiments can provide strong results for the behavior of the networks between their n vantage points (typically less than 50), and link coverage grows as $O(n^2)$ for small n, but edge coverage is only $O(n)$. Without probing outside the mesh, however, these approaches ultimately study only a small fraction of the entire Internet. Other methods of active probing, and our work, aim to provide complete coverage.

In early work, Paxson reports routing failures in about 1.5%–3.3% of trials [35]. A more recent work, the RON system reports 21 "path-hours" of complete or partial outages out of a total of 6825 path-hours, a 0.31% outage rate [1]. Feamster et al. measure Internet path failures with $n = 31$, and correlate with BGP messages for causes [17]. They find that most failures are short (under 15 minutes) and discuss the relationship between path failures and BGP messages. As with their work, we validate our findings using control plane data.

The instrumentation in these systems can often isolate locations of problems, such as SCORE (Kompella et al. [26]); work that complements ours.

Rather than a mesh, PlanetSeer studies traffic from 7–12k end-users to a network of 120 nodes to track path outages [47]. They report that their larger population identifies more anomalies than prior work; we expect our edge coverage of 2.5M blocks will be broader still. In addition, their measurements occur only on clients; they miss outages from already disconnected clients.

Client support in these studies allows better fault diagnosis than our work. Our work complements theirs by providing much larger coverage (2.5M /24 blocks, a large fraction of the Internet edge), rather than "only" hundreds or thousands; and supporting regular, centrally driven measurement, rather than client-driven measurements that undercount outages.

2.4 Passive Data Analysis

Recent works by Dainotti et al. do an in-depth analysis of Internet outages caused by political censorship [13, 14]. Their main focus is the Egypt and Libya outages in 2011, using a novel approach that combines observations from both control-plane (BGP logs) and data-plane sources (backscatter traffic at UCSD network telescope and active probing data from Ark). They focus on the use of multiple passive data sources; they find their source of active probes is of limited use because it probes each /24 every three days. We instead show that a single PC can actively probe all visible and responsive /24 blocks every 11 minutes (§4), suggesting active probing can provide complement to them.

Above the network layer, other systems have looked at system- and user-level logs to determine outages. For example, UCSD researchers have done careful studies of "low-quality" data sources (including router configurations, e-mail and syslog messages), to discover characteristics and reasons of failures in the CENIC network [42]. Such log analysis requires collaboration with the monitored networks, and so their study focuses on a single ISP. In contrast, we use active probing that can be done independent of the target.

2.5 Origins of Routing Instability

BGP centralization of otherwise distributed routing information makes it an attractive source of data for outage analysis. Prior work has used the AS path to study where outages originate. Chang et al. cluster BGP path changes into events, both temporally and topologically [7]. They also provide insights on how to infer where network events happen. Feldmann et al. identify ASes responsible for Internet routing instabilities using time, views and prefixes [18]. They report that most routing instabilities are caused by a single AS or a session between two ASes. (Chang et al. make similar conclusions [7]). They also propose useful insights on hazards in identifying instability originators. We develop conceptually similar clustering methods, but based on data-plane observations. Our active probing approach finds many large Internet outages that cut across multiple ASes, and also detects outages in edge networks that use default routing.

Network tomography uses coordinated end-to-end probes to detect the specific location of network failures [12,15, 22]. We also identify outages near our vantage points to correct for errors (§6.3). However, our work is in a different domain, as our focus is to analyze the end-to-end reachability of the whole Internet.

3. METHODOLOGY

Our method for outage detection begins with active probing, followed by outage identification in individual blocks, visualization, and correlation into events.

3.1 Active Probing of Address Blocks

We collect data with active probing, building on our approach developed to study the Internet address space [20]. A brief review of this collection method and data normalization follows. In §4 we extend raw collection into a system optimized for outage detection.

Reviewing Address-Space Probing: Our approach begins with active probing of some or all addresses in some or all analyzable /24 address blocks in the IPv4 address space. We probe each block with ICMP pings (echo requests) at 11 minute intervals for one to 14 days. Responses are classified into four broad categories: positive (*echo reply*), negative indicating network is unreachable (for example, *destination unreachable*), other negative replies (we interpret these as a reachable network), and non-response. We have two probing configurations: Internet address surveys probe all addresses in about 22,000 /24 blocks (data available [20] and reviewed in §5.1), while the operational outage observation system probes 20 addresses in 2.5M /24 blocks (§4).

Our probing rate is high compared to some prior probing systems. When we probe all addresses in a /24, incoming probe traffic to each /24 arrives at a rate of one packet every 2.5 s. In operation, we get about three inquiries about probing per month, either directly to the ISP or through information on a web server on the probers. Many requests are satisfied when they understand our research, but can be added to a do-not-probe blacklist on request. Our operational system (§4) probes many more blocks, but at a rate of one packet every 32 s, actually drawing fewer complaints.

Our outage detection applies only to blocks where 10% of addresses respond (§3.2). Based on Internet-wide censuses, about 17% of /24 blocks meet this criteria [20]. Our results therefore exclude sparsely populated blocks, but do reflect on a diverse set of Internet users whose firewalls admit ICMP, including home users, server farms, universities, and some businesses. Although we provide no information about the non-responsive Internet, this limitation is shared by other forms of active probing, and our coverage is actually 14% better than Hubble in §5.6.2.

Normalizing survey data: Probes are spread out in time and responses return with varying delays in the raw data. In this paper we simplify the survey data by mapping probe records into *rounds*, where each round is 11 minutes long. We identify rounds by index i, with N_r total rounds in a dataset (thus $i \in [1 \,..\, N_r]$).

We correct two errors that occur in mapping observations to rounds: sometimes a round is missing an observation, and occasionally we see duplicate responses in that round. Our collection software is not perfectly synchronized to 11 minute rounds, but takes on average 11 minutes and 3 seconds. (We intentionally chose to correct for minor drift rather than guarantee perfect synchronization over days of continuous operation.) Because this interval is not exactly 11 minutes, for each individual IP address, about one round in 220 has no observation. We detect such holes and fill them by extrapolating from the previous observation. In addition, we sometimes get multiple observations per round for a single target. About 3% of our observations have duplicate results, usually a timeout (non-response) followed by a negative response (an error code). These duplicates are rare, and somewhat non-uniformly distributed (for example, about 6% of blocks have over 100 addresses each reporting duplicates, but most blocks have no duplicates). When we get duplicate responses, we keep the most recent observation, thus the negative response usually overrides the timeout.

Finally, we observe that the process of associating the IP address of an ICMP reply with its request is not perfect. Multi-homed machines sometimes reply with an address of an interface other than the one which was targeted, this is known as IP address aliasing in topology discovery (as described in early work [19] and recent surveys [24]). Since we know all the addresses we probe, we discard responses from unprobed targets (about 1.4% of replies).

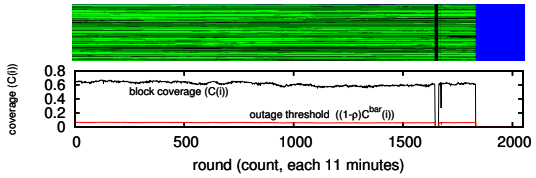


Figure 1: *Top*: probe responses for one /24 block. Green: positive response; black: no response; blue: not probed (after round 1825). *Bottom*: block coverage and outage thresholds per round. Dataset: Survey S_{30w}.

3.2 Probes to Outages

From a series of probe records organized into rounds, we next identify potential outages when we see a sharp drop and increase in overall responsiveness of the block.

Our system begins with observations of individual addresses. Let $r_j(i)$ be 1 if there is a reply for the address j in the block at round i, and 0 if there is no reply, or the negative response is network or host unreachable.

$$r_j(i) = \begin{cases} 1, & \textit{responsive} \\ 0, & \textit{otherwise} \end{cases}$$

Figure 1 shows a graphical representation of $r_j(i)$: each green dot indicates a positive response, while black dots are non-responsive (the blue area on the right is after the survey ends). In this block many addresses are responsive or non-responsive for long periods, as shown by long, horizontal green or black lines, but there is some churn as machines come and go.

The *coverage* of a block, at round i, is defined as:

$C(i) = \frac{1}{N_s} \sum_{j=1}^{N_s} r_j(i)$.

(Where N_s is the number of IP addresses that are probed in the block, either 256, or 20 with sampling §4.2.) $C(i)$ is a timeseries ($i \in [1 \,..\, N_r]$), for block responsiveness across the entire observation period.

A severe drop and later increase in $C(i)$ indicates an outage for the block. The bottom of Figure 1 shows $C(i)$ drops to zero for rounds 1640 to 1654, an outage that shows as a black, vertical band in the top panel.

Algorithm 1 formalizes our definition of "a severe drop": we keep a running average of coverage over window w (default: 2 rounds or 22 minutes) and watch for changes of $C(i)$ by more than a threshold ρ (default: 0.9). In a few cases $C(i)$ changes gradually rather than suddenly, or a sudden change is blurred because our observations are spread over 11 minutes. Therefore, for robustness of algorithm, we compare $C(i)$ against both the current running average, and the previous round's running average. The result of this algorithm is a list of outages and a binary-valued timeseries $\Omega(\cdot)$, indicating when the block is down ($\Omega(i) = 1$) or up (0). For succinctness, we don't show other special cases in Algorithm 1 (such as consecutive downs/ups, where we mark earliest as down and latest as up), but we handle

Algorithm 1 Outage detection for a block

Input: $C(i)$: timeseries of coverage, N_r: number of rounds
Output: L: list of outage (start, end) time tuples
$\Omega(i)$: binary timeseries of block down/up information.
Parameters: w: number of rounds to look back, ρ: drop/increase percent to trigger outage start/end

```
L = ∅, Ĉ = 0
Ω(i) = 0, i ∈ [1..N_r]
for all i ∈ [w + 1..N_r] do
  Ĉ' = Ĉ // previous running average
  Ĉ = (1/w) Σ_{j=i−w}^{i−1} C(j) // current running average
  if C(i) < (1 − ρ)Ĉ or C(i) < (1 − ρ)Ĉ' then
    // severe drop ⇒ outage start
    last_outage_start ← i
  else if Ĉ < (1 − ρ)C(i) or Ĉ' < (1 − ρ)C(i) then
    // severe increase ⇒ outage end
    L = L ∪ {(last_outage_start, i)}
    for all j ∈ [last_outage_start..i] do
      Ω(j) = 1
    end for
  end if
end for
return L, Ω
```

such cases properly in our implementation. Also, we report outage as long as $C(i)$ is 90% lower than previous rounds, even if $C(i) > 0$ in some cases.

Because this algorithm detects changes in $C(\cdot)$, it only works for blocks where a moderate number of addresses respond. We typically require around $\alpha = 0.1$ of all addresses (10% or 25 address per /24), in a block to respond, averaged over the entire survey ($\bar{C} = (1/N_r) \sum_i C(i) \geq 0.1$), otherwise we ignore the block as being too sparse. In §3.6 we review values of α and conclude that $\alpha = 0.1$ is reasonable. Table 1 shows how many blocks are analyzable for Survey S_{30w} (the 30th survey taken, in the U.S. west coast). In our operational system (§4), we pre-screen blocks, discard sparse blocks (less than 25 responders), probe only the 20 addresses most likely to respond; we therefore omit the α-check in this case.

3.3 Visualizing Outages

With the above algorithm to find block-level outages, we next develop a simple clustering algorithm to group block-level outages in two dimensions: time and space. We use this algorithm for visualization only; in the next section we show a second clustering algorithm that correlates outages to network events.

Our clustering algorithm (Algorithm 2) orders blocks based on Hamming distance. For blocks m and n, with binary-valued outage timeseries $\Omega_m(i)$ and $\Omega_n(i)$, we

category	blocks	percentage	
all IPv4 addresses	16,777,216		
non-allocated	1,709,312		
special (multicast, private, etc.)	2,293,760		
allocated, public, unicast	12,774,144	100%	
non-responsive	10,490,902	82%	
responsive	2,283,242	18%	100%
probed	*22,381*		*1%*
too sparse, $\bar{C} < \alpha$	*11,752*		*0.5%*
analyzable, $\bar{C} \geq \alpha$	**10,629**		**0.5%**

Table 1: Subsetting for blocks that are *probed* and **analyzable** ($\bar{C} \geq 0.1$), for Survey S_{30w}. Measurements are in numbers of /24 blocks. The percentages are shown on a per-column basis (e.g., responsive blocks are 18% of allocated,public and unicast blocks).

Algorithm 2 Clustering of blocks for visualization

Input: A: the set of all blocks in a survey, with outage information

Output: B: list of survey blocks, ordered by distance

```
start with block m ∈ A with smallest Σ_{i=1}^{N_r} Ω_m(i)
(number of rounds down)
A = A \ {m}
B.append(m)
while A ≠ ∅ do
  for all n, s.t. d_h(m, n) = 0 do
    A = A \ {n}
    B.append(n)
  end for
  // pick the next most similar block:
  find m' s.t. d_h(m, m') ≤ d_h(m, n) ∀n ∈ A
  A = A \ {m'}
  B.append(m')
  m = m'
end while
return B
```

define distance:

$d_h(m,n) = \sum_{i=1}^{N_r} \Omega_m(i) \oplus \Omega_n(i)$.

Perfect temporal correlation occurs if $d_h(m,n) = 0$.

Figure 2 shows the result of visualization clustering for Survey S_{38c}. The x-axis is time, each row shows the Ω_j downtime for a different /24 block j. Due to space, we plot only the 500 blocks with most outages. Color is keyed to the country to whom each block is allocated.

We discuss the details of this survey in §5, but there are two clusters of blocks that have near-identical outage end times. The cluster labeled (a) covers 19 /24s that are down for the first third of the survey; it corresponds to the Feb. 2011 Egyptian Internet shutdown. The cluster labeled (b) covers 21 /24 blocks for a slightly longer duration; it is an outage in Australia concurrent with flooding in the eastern coast.

3.4 Outages to Correlated Events

Next we use block outage information to discover network events; we use these events later in §5 to relate the outages we see to ground truth based on routing and news. While visualization is helpful, Algorithm 2 overconstrains clustering since each block can be adjacent to only two others.

We therefore develop a second clustering algorithm that relaxes this constraint, instead of grouping blocks, we group individual block-level outages into network-wide events. We identify events from similar start- and end-times of outages. Given two outages o and p, each having a start round $s(\cdot)$ and end round $e(\cdot)$, we measure their distances d_e:

$d_e(o,p) = |s(o) - s(p)| + |e(o) - e(p)|$

Outages that occur at exactly the same time have $d_e(o,p) = 0$. Clusters can be formed by grouping all outages that occur at similar times. Since routing events often require some time to propagate [27], and outages may occur right on a round edge, we consider outages with small distance (less than a parameter θ) to be part of the same event. This approach may fail if there are two unrelated events with similar timing, but we believe that timing alone is often sufficient to correlate larger events in today's Internet, provided we use a conservative θ. Currently we set $\theta = 2$ rounds (22 minutes). We have also studied much larger $\theta = 10$ (110 minutes), showing similar results, although less strict matching aggregates many more small events, see §6.2. This is formalized in Algorithm 3.

Discussion: For simplicity and efficiency, we use greedy $O(n^2)$ clustering algorithms. (Algorithms 2 and 3). We considered other standard clustering algorithms, including k-means and hierarchical agglomerative clustering. The k-means algorithm is not suited for our problem, because k needs to be pre-selected as the number of clusters, which is not known beforehand. We don't choose hierarchical agglomerative clustering for

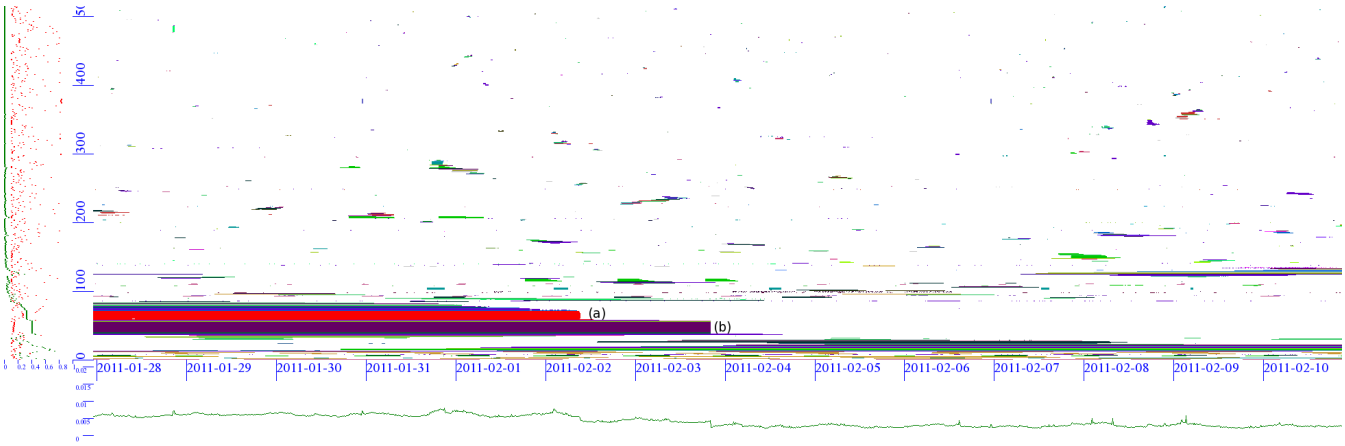


Figure 2: The 500 largest outages of S_{38c}, x axis: time, y axis: address space (blocks). Colors represent countries. Subgraphs on X and Y axis show marginal distributions (green line) and overall block responsiveness (red dots).

Algorithm 3 Finding correlated events

Input: O: the set of all outages in a survey
Output: E: the set of network outage events, each containing one or more outages
Parameters: θ: the threshold to decide if two outages belong to same event

```
while O ≠ ∅ do
    find first occurring outage o ∈ O
    e = {p : ∀p ∈ O, s.t. d_e(o, p) ≤ θ }
    O = O \ e
    E = E ∪ {e}
end while
return E
```

efficiency reasons, because it has a time complexity of $O(n^3)$ and is not suitable for a large n (especially for the operational system in §4).

3.5 Outages to Internet Availability

To evaluate outages over the Internet as a whole, we next define statistical measures of Internet *availability*, how many and how long blocks are unreachable.

As shown in Figure 3, some network events like event (c) affect many blocks for a short period (here, about 20 minutes), while others like (d) and (e) affect fewer blocks but for longer periods of time (here 2 to 3 hours). We discuss these events in detail in §5.2, but they suggest that *marginal distributions* of outages would be useful to capture this space-time behavior.

Given N_b blocks and N_r rounds in a survey, we can compute the time- and space-specific sums:

$$\bar{\Omega}_I(i) = \sum_{b=1}^{N_b} \Omega_b(i) \qquad \bar{\Omega}_B(b) = \sum_{i=1}^{N_r} \Omega_b(i)$$

We normalize $\bar{\Omega}_I(i)$ by N_b and $\bar{\Omega}_B(b)$ by N_r in the subgraphs of our outage plots (such as Figure 2), and report absolute values in §6.3.

Finally, we define the overall outage level as the fraction of time and space that was out over all observations: $\bar{\Omega} = (N_b N_r)^{-1} \sum_{i=1}^{N_r} \sum_{b=1}^{N_b} \Omega_b(i)$

3.6 Parameter Discussion

We next discuss the parameters of our approach to evaluate how sensitive the results are to their values.

We use a window w (default: 2 rounds) to determine edges of outages. A large w is not feasible because most outages are short (§6.2). We studied different w values from 1 to 5 rounds, and found that the numbers of up/down decisions ($\Omega(\cdot)$) differed by only 0.3%, confirming our choice of $w = 2$ is reasonable.

The parameter ρ (default: 0.9) is the fraction of addresses must go dark to indicate an outage. To evaluate the effect of ρ, we consider an extreme strategy *any* as ground truth, where we probe all addresses, but consider the block up if any single address responds. We choose $\rho < 1.0$ because requiring a "perfect" outage allows a single router to indicate a block is up even if all hosts are down. However, the difference in accuracy is less than 0.1% (details in [36]). For ρ values of 0.5 to 0.9, outage estimates are all accurate (more than 99.8%), differing by less than 0.1%. We select $\rho = 0.9$ as a balance between accuracy and conservativeness, when declaring an outage.

We define outage $\Omega(\cdot)$ as a property of an entire /24 block, implying the entire /24 is used and routed consistently. About 76% of addresses are in consistently used /24s [6]; study of sub-/24 outages is future work.

We use α to identify blocks as too sparse to classify because of few responding addresses. A very small α is not possible because $\lfloor \alpha(1-\rho)N_s \rfloor$ must be more than 1 (§3.2), and large enough to be robust to packet loss. A large α would disqualify many blocks (§4.3). An α of 0.1, meaning that on average 25 hosts in a fully probed block are responsive, is a good balance between probing rate (§4) and accuracy (§5.6.1).

The choice of an 11-minute probing interval limits the precision of our estimates of outage times. We se-

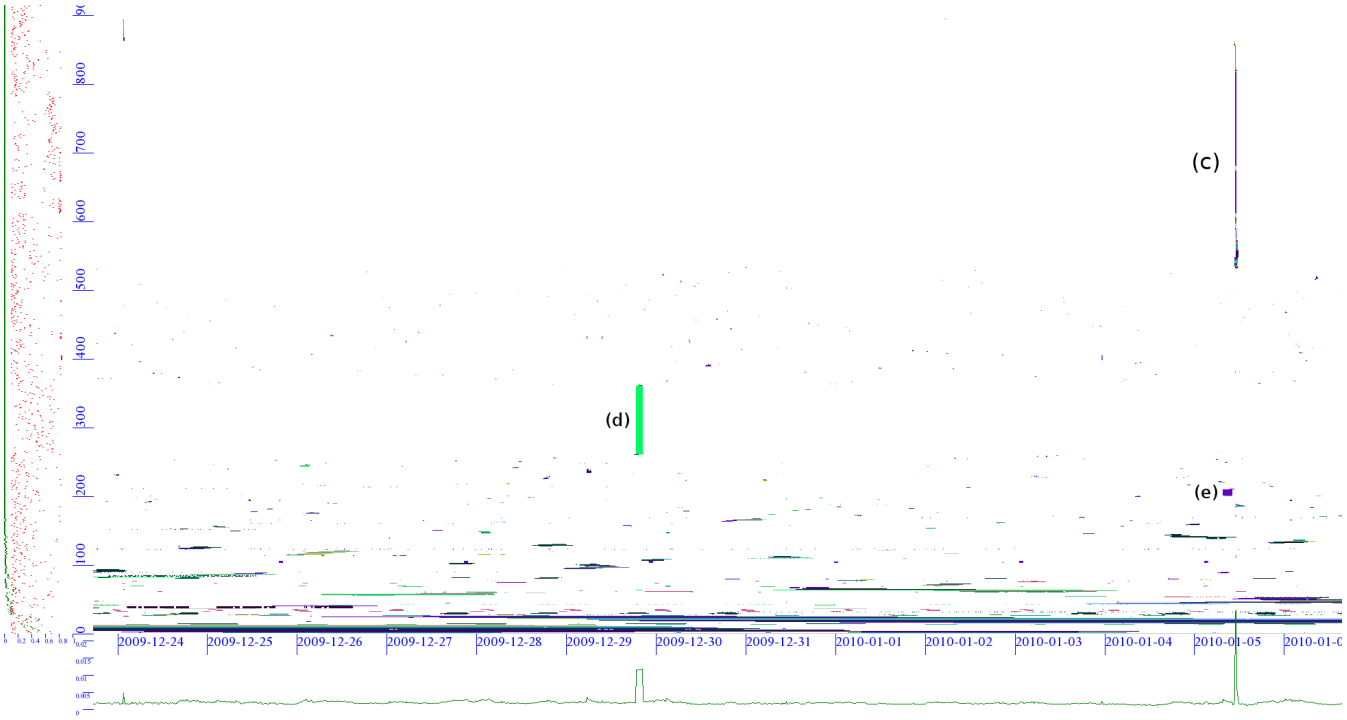


Figure 3: The 900 largest outages of S_{30w}, x axis: time, y axis: address space (blocks). Colors represent countries. Subgraphs on X and Y axis show marginal distributions (green line) and overall block responsiveness (red dots).

lected this probe frequency to match that used in public datasets [20], as it provides a reasonable tradeoff between precision and traffic, and because our analysis is greatly simplified by a fixed probing interval. Our choice limits the probing rate of the target networks to no more than one probe every 2.5 s, when all addresses are probed.

4. BUILDING AN OPERATIONAL SYSTEM

Much of our analysis uses complete probing: survey data probes all addresses in each /24 block every 11 minutes. This traffic is modest at the targets (each /24 receives one probe every 2.5s), and for 22k blocks the prober sends around 6k probes/s. However, covering the entire public IPv4 space would be expensive: 4.8M probes/s at the source, requiring more than 250 cores and 2.5Gb/s traffic. We next describe our operational probing system. We identify plausible probing rates for targets and prober, and develop optimizations to reduce the traffic at the target and the load on the prober.

4.1 Bounding Probing Traffic

Probing rates trade traffic against accuracy, so we first identify reasonable rates for the prober and target.

At the target, probing all addresses in each /24 block every 11 minutes implies 0.39 probes/s per block. To put this traffic in perspective, a typical /24 block receives 0.56 to 0.91 probes/s as "background radiation" (22 to 35 billion probes per week per /8 block [45], ignoring unusual targets like 1.2.3.4). Full block probing therefore imposes a noticeable burden on the target: about 50% more background traffic. We therefore probe only 8% of the addresses in each block, cutting per-block incoming traffic to a only 4–7% of background.

At the prober, outgoing probes are constrained by bandwidth, CPU, and memory, as we track probes awaiting responses. Of these, CPU is the largest constraint, since we require 75k probes/s, about 40Mb/s outgoing traffic, and each open request requires only 104 bytes of memory. We show below that we can reduce the number of target blocks to allow a single, modest 4-core server to probe all of the analyzable IPv4 Internet.

4.2 Sampling Addresses in Blocks

While probing all addresses gives a perfect view of the block, much of that traffic is redundant if one assumes outages affect all or none of the block. (Prior work suggests that about 76% of addresses are managed as /24-size blocks or larger [6], so this assumption usually holds.) Some redundancy is important to avoid interpreting individual failure as outage for the entire block, therefore next we evaluate sampling k of these addresses ($k \ll 256$).

Sampling reduces the probing rate, but also accuracy. To maximize the benefit of probing we wish to probe the addresses in each block that are most likely to respond, since only they can indicate when the block is out. This problem is a generalization of hitlist detection, which selects a single representative address for

each block [16]. Instead we want the k-most likely to respond. We use public datasets derived for hitlist generation, consuming two years of full IPv4 census data to find the k-sample addresses customized for each /24 block. We evaluate the effects of sampling in §5.6.1, showing that $k = 20$ provides good accuracy.

4.3 Reducing the Number of Target Blocks

To detect outages successfully, several addresses in a block must reply to probes. Many blocks in the IPv4 address space do not respond to pings at all [20]; they are firewalled, not routed on the public Internet, or not occupied. Many more blocks have a few addresses that respond, but not enough for our threshold ($\alpha = 0.1$ in §3.2). Therefore we discard such non-analyzable blocks.

To evaluate how many /24 blocks respond and meet our criteria of analyzable, we looked at a census of all IPv4 addresses [20] taken at the same time as S_{40w}. There were 14.4M /24 blocks allocated, but only 4.0M (28%) had any responses, and only 2.5M (17%) are analyzable, meeting our threshold of 25 ($\lfloor 256\alpha \rfloor$) or more responders. In summary, we can cut our aggregate probe rate by a factor of about 75 by avoiding non-analyzable blocks and downsampling in each block.

4.4 Our Prototype System

Our prototype probing system employs both of these optimizations, probing 20 samples in each of about 2.5M /24 blocks (75k probes/s) to observe the entire analyzable IPv4 Internet. This target population requires about 40 Mb/s outgoing network traffic and sees about 27 Mb/s return traffic. Our core prober is prior work (from [20]), but preparation, analysis, and optimizations so one host can cover the Internet are new.

Probe Preparation: Before beginning a probing run, we must generate the list of target blocks and sampled addresses. (In regular use, we would redo this list for each new census.) The input for this process is the most recent *IPv4 Response History* dataset, containing the estimates of how likely each and every IPv4 address is to respond [16] based on approximately two years of rolling IPv4 censuses [20]. We extract the k-sample addresses for each /24 block using a Hadoop-based Map/Reduce job. The output of this step is a list of IP addresses for each viable block. The address list is written in a pseudo-random order, so probes to each block are spread out over each round (to avoid ICMP rate limiting at target blocks), and probe order is different for each block. (We use probing order described previously [20], similar to that of Leonard&Loguinov [29].)

Active Probing: We use a custom high-performance prober, to pace probes across the 11-minute round duration, send many probes without waiting, and track their progress until they reply or timeout in 3s. It associates replies with requests based on the reply address (80.6% of the time), the contents of the reflected header (0.5%), or it logs the apparently erroneous reply (18.9%). We run four instances of the prober in parallel on a single computer, each processing one quarter of the targets.

Response Analysis: We analyze the responses when collection completes, or periodically for on-going collection. We process the data with three Map/Reduce jobs: first we convert raw responses from each address into discrete records by 11-minute rounds; then we group these records by common prefix for each /24 block; finally we compute outages for each block (Algorithm 1). We also cluster and plot outages for further analysis.

From 2011-09-28 T22:36 +0000, we have taken a 24-hour probe of sampled addresses for entire analyzable IPv4 Internet. That observation of 2.5M blocks includes about 6.5 billion records and 56GB of compressed data. By comparison, a two-week survey of 22,000 blocks consists of about ten billion records and 70GB of compressed data. While we have not tried to optimize our analysis code, we can turn observations into clustered events in about 80 minutes for a survey on our cluster.

Performance: In operation we run four parallel probers (4-way parallelism), each a separate process on a CPU core, probing a separate part of address space. We find each core can sustain 19k probes/s and conclude that a single, modest 4-core server can probe all of the analyzable IPv4 Internet (see Appendix C for details).

Data Availability: Our input data and results are available on request at `http://www.isi.edu/ant/traces/index.html`.

5. VALIDATING OUR APPROACH

We next validate our approach, starting with case studies, then consider unbiased random cases and stability over time and location. Finally, we compare our accuracy to prior approaches.

5.1 Validating Data Sources and Methodology

While our current operational system probes the analyzable Internet, to validate our approach we turn to survey data collected over the last two years. We use survey data here to provide an upper bound on what full probing can determine, and because our optimized system was only completed in Sep. 2011; we show in §6.1 that our optimized system is consistent with complete probing. Our goal is to confirm our observations by verifying against *real-world events*: public archives of BGP routing information and, for large events, public news sources. We next summarize our datasets, how we use BGP, and how we associate an event to specific Autonomous Systems.

Datasets: We use 35 public Internet survey datasets collected from Nov. 2009 to Dec. 2011 [43] (S_{29w} through S_{40w}, see Appendix A for full list). Each dataset represents two weeks of probing; data is taken from three

locations (Marina del Rey, California; Ft. Collins, Colorado; and Keio University, Tokyo, Japan). Each dataset probes all addresses in about 22,370 /24 blocks where three-quarters of blocks are chosen randomly from responsive blocks, while one quarter selected based on block-level statistics [20]. Since some blocks are selected non-randomly, §5.5 evaluates bias, finding we slightly underestimate outage rates.

We find that 45–52% of blocks in these datasets provide enough coverage to support analysis ($\bar{C} \geq 0.1$). Of these datasets, most validation uses S_{30w} (started 2009-12-23), with additional case studies drawn from S_{38w} (2011-01-12), S_{38c} (2011-01-27), S_{39w} (2011-02-20) and S_{39c} (2011-03-08).

We gather BGP route updates from RouteViews [34], and BGP feeds at our probing sites using BGPmon [46].

Relating events and routing updates in time: To find routing updates relevant to a network event, we search BGP archives near the event's start and end times for messages concerning destination prefixes that become unreachable. We search within 120 minutes of these times, a loose bound as our outage detection precision is only ±11 minutes, and routing changes can take minutes to converge. We expect to see relevant withdraw messages before event e and announce messages after e. Finding both, we claim that e is *fully validated*, while with just one we claim *partial validation*.

Relating events and routing updates in space: Although the above approach detects outages that happen at the destination, we find many outages occur in the middle of the Internet. Narrowing our search to just destination prefixes therefore overly constrains our search. When our temporal search fails to identify a routing problem, we broaden our search to all ASes on the path, as done by Chang et al. [7] and Feldmann et al. [18]. We generate an AS path for the destination prefix by searching in RouteViews BGP snapshots. We then search for BGP withdraw and announce messages around the same time as the start and end of our network event. Often the destination search found an announce message; in that case we look here for withdraw messages for an intermediate AS.

Searching intermediate ASes has two disadvantages. First, the search space is much larger than just considering the destination prefixes. Second, RouteViews BGP snapshots are taken every two hours, so we must widen our search to two hours.

5.2 Network Event Case Studies

We begin by considering three cases where the root cause made global news, then outages near our collection points, and finally three smaller events. These events are larger than the median size outages we detect. We make no claims that these events are representative of the Internet in general, only that they demonstrate how events found by our tools relate to external observations. In the next section we validate a random sample of events to complement these anecdotes.

Jan. 2011 Internet Outage: Beginning 2011-01-25 the Egyptian people began a series of protests that resulted in the resignation of the Mubarak government by 2011-02-11. In the middle of this period, the government shut down Egypt's external Internet connections.

Our S_{38c} began 2011-01-27 T23:07 +0000, just missing the beginning of the Egyptian network shutdown, and observed the restoration of network service around 2011-02-02 T09:28 +0000. Our survey covered 19 responsive /24 blocks in the Egyptian Internet, marked (a) in Figure 2. We can confirm our observations with widespread news coverage in the popular press [41], and network details in more technical discussions [9, 10]. Analysis of BGP data shows withdraws before and announces after the event, consistent with our timing. All Egyptian ASes we probed were out, including AS8452, AS24835, and AS24863. We conclude that our approach correctly observed the Egyptian outage.

Feb. 2011 Libyan Outage We also examined the Libyan outages 2011-02-18 to -22 [11]. This period was covered by S_{38c}, but our survey contains only one Libyan block, and coverage for that block was too low (about 4 addresses) for us to track outages. Our requirement for blocks with moderate coverage, combined with measuring only a sample of the Internet and Libya's small Internet footprint (only 1168 /24 blocks as of Mar. 2011 [44]) shows that we sometimes miss outages.

Feb. 2011 Australian Outage: We also observe a significant Australian outage in S_{38c}. Marked (b) in Figure 2, by our observations this outage involved about as many blocks as the Egyptian outage. We can partially validate our outage with BGP, but its root cause is somewhat unclear. We are able to locate these blocks in the east coast of Australia, including Sydney and Brisbane. Private communications [2] and the AusNOG mailing list [3] suggest this outage may be related to mid-January flooding in eastern Australia. However, our survey begins on 2011-01-27, so we only know the outage's end date. The recovery of the network seems consistent with news reports about telecommunications repairs [39]. Our observations suggest that this Australian outage was about *as large and long-lasting* as the Egyptian outage, yet the Egyptian Internet outage made global news while the Australian outage got little discussion. The Egyptian outage was more newsworthy both because of the political significance, and because it represented nearly all Egyptian traffic. Australia, by comparison, has eight times more allocated IPv4 addresses than Egypt, so though the Australian outage may be as large as the Egyptian one, it does not have the same country-wide impact. We believe this example shows the importance of our methodology to *quantify*

the size and duration of network outages.

March 2011 Japanese Earthquake: In survey S_{39c}, we observe a Japanese Internet outage, as shown in Figure 4 marked (f). This event is confirmed as an undersea cable outage caused by the Tōhoku Japanese earthquake 2011-03-11 [32]. Unlike most other outages we observe, both the start and recovery from this outage vary in time. For most blocks, the outage begins at the exact time of the earthquake, but for some it occurs two hours later. Recovery for most blocks occurs within ten hours, but a few remain down for several days.

Local Outages: In addition to outages in the Internet, they also happen near our monitors. (We watch for such outages in our data, and confirm with local network operations.) Survey S_{39w} shows two such events. In Figure 5, event (h) was planned maintenance in our server room; the blue color indicates absence of data. Event (i) was a second planned power outage that took down a router near our survey machines although probes continued running. Both of these events span all probed blocks, although Figure 5 shows only 500 of the blocks. Finally, event (g) is due to temporary firewalling of our probes by our university due to a mis-communication.

These examples show that our methods have some ability to distinguish local from distant outages. They also revealed an interaction of our probing with Linux iptables. In event (i), the number of active connections in iptables overflowed. Such overflow produces random ICMP network unreachable error replies at the probing host. We filter these errors from our prior data, and have now disabled ICMP connection tracking.

Smaller Events: Finally, we explore three small events in survey S_{30w} as examples of "typical" network outages. These events are shown in Figure 3. Although we don't find evidence in the NANOG mailing list, BGP messages do confirm two of them.

Verizon outage 2010-01-05 T11:03 +0000: In Figure 3, event (c) is a short outage (about 22 minutes) affecting about 331 /24 blocks. Many of these destinations belong to AS19262, a Verizon AS. Examination of RouteViews BGP archives confirms this event. Examination of the AS-paths of affected blocks suggests that the outage occurred because of a problem at AS701, another Verizon AS, present in the path of all but 0.6% of destinations. It also confirms the duration, with the BGP withdraw-to-announce time of about 20 minutes.

AT&T/Comcast 2010-01-05 T07:34 +0000: In Figure 3, event (e) is a 165 minute outage affecting 12 blocks. Again, we confirmed this outage in RouteViews BGP archives. The affected destinations were AS7132 (AT&T) and AS7922 (Comcast). Routing archives confirm withdraws and returns of these routes, and AS-paths suggest the root cause was in AS7018 (AT&T WorldNet), likely upstream of the destinations.

Mexico outage 2010-12-29 T18:36 +0000: The event labeled (d) in Figure 3 corresponds to a large number of destinations in AS8151, a Mexican ISP (Uninet S.A. de C.V.). The event is fairly large and long: 105 blocks for 120 minutes. We were unsuccessful in identifying the root cause of this outage in RouteViews data. This survey pre-dates our local BGP feed, and all RouteViews BGP archives are several ASes from our probing site, suggesting the outage may have been visible to us but not seen at the RouteViews monitors, or that some of these blocks may be using default routing as described by Bush et al. [5].

valid.	with.	ann.	count		outage sizes
no	—	—	31	(62%)	1 to 57, median 4
partial	Yes	—	1	(2%)	24
partial	—	Yes	10	(20%)	1 to 27, median 15
yes	Yes	Yes	8	(16%)	1 to 697, median 21
			50	(100%)	

Table 2: Validation of algorithm with counts of missing (—) or found (Yes) withdraw and announce messages, for randomly selected events from Survey S_{40w}. Counts in events; sizes in blocks.

5.3 Validation of Randomly Selected Events

Our outage case studies in the prior section were selected because of their importance and so are biased towards larger events. To provide a more careful study of the validity of our approach, we randomly pick 50 events from a total of 1295 events in Survey S_{40w} and attempt to confirm each using BGP information (§5.1).

Table 2 summarizes our results. We are able to fully or partially confirm 38% of the cases by finding either corresponding BGP withdrawal or announcement messages. Randomly selected events are often small (as confirmed in §6.2), and it is easier to verify large events. One possible reason smaller events do not appear in the control plane is that smaller networks more often use default routing. Bush et al. describe how default routing can result in "reachability without visibility", as addresses may be reachable without visibility to the BGP control plane [5]. Our results are consistent with a corollary, "outages without visibility", since outages in default-routed blocks do not appear in BGP. We therefore claim that 38% represents *incompleteness of BGP* and not our detection algorithm; we next use controlled outages to support this hypothesis.

5.4 Validation of Controlled Outages

Evaluation of random events show what we detect is true, but it is silent about what we miss. We next show our system can detect *all* outages of sufficient duration.

To provide a controlled experiment, we extract probes sent from California to five known /24 blocks in Colorado from our analyzable Internet experiment (§6.1). Network operators confirm these blocks had no outages on that day. We use real probing data to capture the

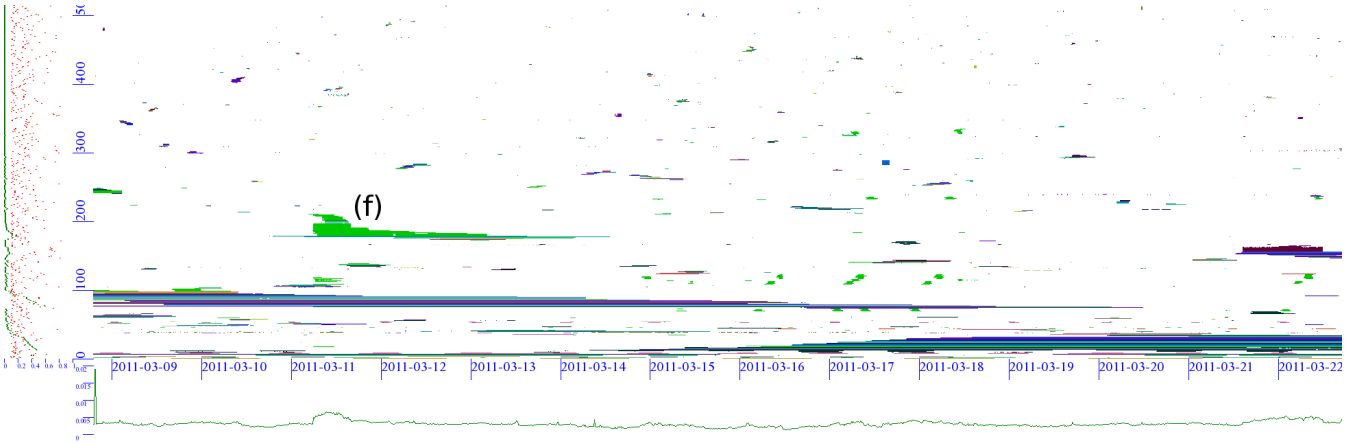


Figure 4: The 500 largest outages in S_{39c}, x axis: time, y axis: address space (blocks). Colors represent countries. Subgraphs on X and Y axis show marginal distributions (green line) and overall block responsiveness (red dots).

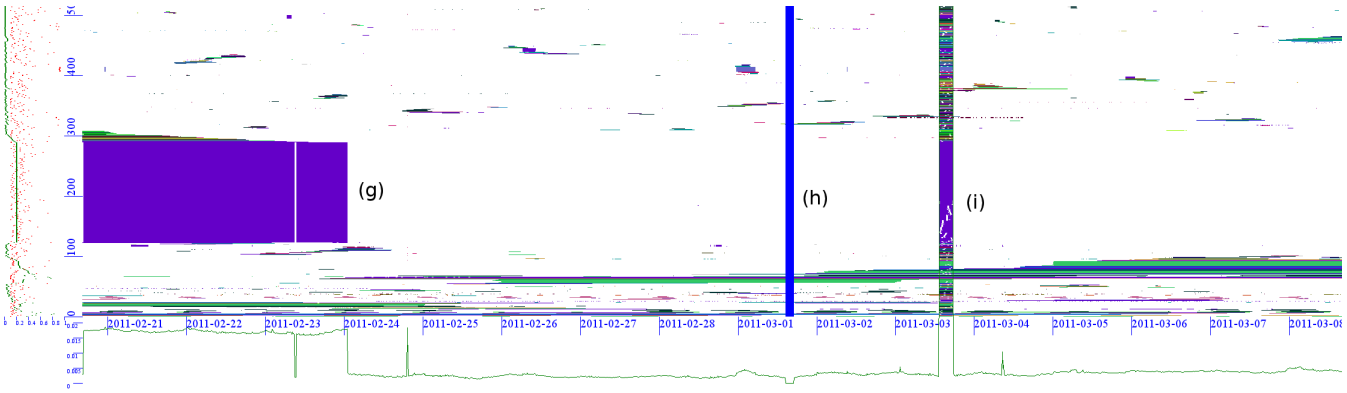


Figure 5: The 500 largest outages in S_{39w}, x axis: time, y axis: address space (blocks). Colors represent countries. Subgraphs on X and Y axis show marginal distributions (green line) and overall block responsiveness (red dots).

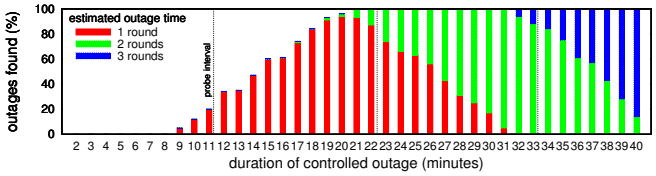


Figure 6: Evaluation of controlled outages on detection (bar height) and estimated duration (color).

noise of packet loss and machine reboots; about 2.1% of our individual probes are negative.

We emulate an outage in each target block by replacing positive responses with negative for one known period. Our emulated outage starts at a random time between 1 hour after collection start and 2 hours before end; start time is thus independent of outage rounds. We vary outage duration, from 1 to 40 minutes in steps of 1 minute, with 100 random times for each step.

Figure 6 shows the percentage of outages we detect for one block as a function of outage duration. All detections for the other blocks (shown in Appendix D) are within 5%. We see that we miss nearly all outages shorter than our probing interval; we space probing out over 11 minutes to be gentle on the target network, creating a low-pass filter over outage observations. As a result, a 5.5 minute outage affecting all addresses appears identical to an 11-minute outage affecting half. We detect *all* outages longer than 21 minutes, and the majority of outages of 15 minutes or longer. Different parameters (§3.6) could adjust sensitivity, but for full-block outages longer than about twice the probe interval, our approach does not falsely declare as available.

Colors in Figure 6 show how long we estimate outages. Due to filtering, we consistently underestimate the duration of each outage by half the probe interval.

5.5 Stability over Locations, Dates and Blocks

We next consider the stability of our results, showing they are independent of prober location and date, and only slightly affected by the survey block select method.

Probing location can affect evaluation results. Should the probing site's first hop ISP be unreliable, we would underestimate overall network reliability. Our probing takes place regularly from three different sites, ISI west (marked "w"), CSU (marked "c") and Keio University (marked "j"), each with several upstream networks.

Figure 7 indicates ISI surveys with open symbols, CSU with filled symbols, Keio University with asterisks, and the analyzable Internet run (§6.1) with inverse open triangle, and it calls out survey location at the top. Visually, it suggests the results are similar regardless of probing site and for many different random samples of targets. Numerically, variation is low: mean outage level ($\bar{\Omega}$) is 0.33% with standard deviation of only 0.1% after local outages are removed. To

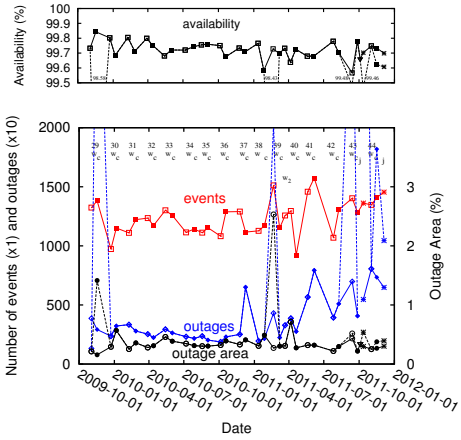


Figure 7: Evaluation over 35 different 2-week surveys, plus our analyzable Internet run. Top shows availability, bottom shows Internet events, outages and outage percentage over time. Outages for S_{29c}, S_{39w}, S_{43w}, S_{43j}, S_{44c}, S_{44j} are 40627, 20362, 35720, 60777, 18189, 10444, omitted from the graph for scale. Dotted lines show statistics without removing local outages.

strengthen this comparison we carried out Student's t-test to evaluate the hypothesis that our estimates of events, outages, and $\bar{\Omega}$ for our sites are equal. The test was unable to reject the hypothesis at 95% confidence, suggesting the sites are statistically similar.

In addition to location, Figure 7 suggests fairly stable results over time, with several exceptions. For example, surveys S_{29c} and S_{39w} each had extended local outages, for about 41 and 4 hours, respectively, shown as dashed lines affecting outage count and $\bar{\Omega}$ (they do not change the event estimate because each outage is mapped to a single network event). After removing local outages, the corrected versions are roughly the same as others.

Only three quarters of blocks in surveys are selected randomly, one quarter are selected to cover a range of network conditions. To evaluate block selection effects, we separate each survey's data into quarters and compared the selected quarter against each of the three randomly chosen quarters. We find that the mean outage rate of the selected quarter is 0.2% (standard deviation 0.078%), while the other three are 0.29% (standard deviation 0.09%). Overall outage estimates from surveys appear slightly more stable (about 0.06% less downtime) than would analysis of a completely random sample. See Appendix B for details.

5.6 Comparing Accuracy with Other Approaches

We probe multiple or all addresses in a block to evaluate outages. Prior work such as Hubble has probed a single address in each block, possibly multiple times [23]. Probing more addresses requires more traffic, but is more robust to probe loss and single-address outages. We next evaluate the effect of sampling k addresses per

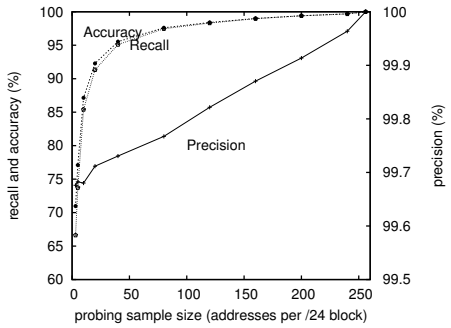


Figure 8: Precision, recall and accuracy as a function of samples per target block. (Neither y-axis starts at 0.)

block, and choice of address for $k = 1$.

To compare alternatives, we evaluate methods A and B in pairs, treating A as a trial and B as truth. Analogous to Type-I and Type-II errors, we define a false availability (fa) as the estimate predicting a reachable block when it should be out, while for a false outage (fo), the estimate predicts out and the truth is reachable. Similarly, we can define true availability (ta) and true outage (to). We then compute standard information retrieval terms: precision ($ta/(ta + fa)$), recall ($ta/(ta+fo)$), and accuracy ($(ta+to)/(ta+to+fa+fo)$).

5.6.1 General Sampling

To evaluate the accuracy of a k-sample, we consider the full probing ($k = 256$) observation of block availability as ground truth (B), then compare our k-sample approximation as an estimate (A).

We evaluate k-samples by downsampling our full data; Figure 8 shows the precision, recall and accuracy for this experiment. Precision is always quite good (over 99.6%), showing it is rare to falsely predict the block as reachable, even when sampling only a few addresses. However, we show below that sampling a single address is less robust than even a few. The best tradeoff of recall and accuracy is for k from 20 to 40, where accuracy off by only 7% (or 4%), but traffic is cut by 92% (or 84%). The errors are mostly due to false outages, claiming the target is down when a more complete measurement would show it as reachable.

5.6.2 Single Address Per Block and Hubble

Hubble, iPlane and most other prior works in outage detection have used a single target to represent an entire /24 block. We next compare our system to these prior systems, quantifying differences in accuracy and coverage. This comparison is difficult because there are several differences in methodology: how many targets to probe per block; probing *which* address or addresses; and use of retries or not. We examine bounds on each of these factors in Table 3, and compare specifically on Hubble's approach (single, .1 address, with-retries) with our approach (top 20 addresses, no-retries). Note that

strategy	single	hitlist	us	Hubble
samples per /24	1	1	20	1
which addresses	.1	top	top	.1
retries	no	no	no	yes
precision	99.97%	99.97%	99.71%	99.98%
recall	56.3%	79.1%	91.3%	61.0%
accuracy	56.4%	79.1%	92.3%	61.1%

Table 3: Comparing accuracy of different strategies used to estimate outage. Dataset: S_{30w} and S_{46c}.

we use a strategy *all* as ground truth, which probes all addresses without retries. We have shown *all* is both complete and accurate in a previous technical report [36].

We discussed the effect of number of targets in §5.6.1, providing a best tradeoff of sampling and accuracy (Figure 8). Prior work on IP hitlists examined the effects of which addresses should be probed [16]. Here we add a comparison of fixed (single) vs. top (hitlist) and we show that top is 22.7% more accurate than only probing a fixed .1 address (79.1% vs. 56.4% in accuracy, Table 3). This shows that probing only the .1 address is not accurate enough for outage detection, and careful selection of which address to probe can improve accuracy significantly.

Using retries should help singleton packet loss, therefore single (.1, no retries) values are underestimates of Hubble. However, considering retries doesn't help with medium-term host failure, such as if the single target is taken down for maintenance. To more accurately evaluate the effect of retries, we run a specific experiment to reproduce Hubble (probing .1 with retries at 2 minute intervals [23]), side-by-side with a recent survey S_{46c}, which we can sample to generate our operational system, and use complete data as ground truth. We find that retries to the same address multiple times is slightly better than no-retries (Hubble vs. single, 4.7% better).

Our side-by-side experiment pulls these factors together, comparing exactly Hubble's configuration (single, .1, with retries) with ours.

We see a 31% improvement in accuracy (us vs. Hubble), consistent with our above bounds.

Probing Rate and Coverage: We have shown that we improve accuracy; in addition, we provide better coverage at about the same aggregate probe rate. Hubble coordinates probes to each block from 30 vantage points, sending 0.5–3 probes/minute on average [23] (ignoring retries). We probe more addresses, but from only one site, thus only 1.8 probes/minute (20 addresses, 1 site, 11 minute cycles).

Finally, our requirement of 20 responsive addresses per block is much stricter than Hubble's one, however, our hitlist-selection is much more flexible. We evaluated coverage using a full census from Jan. 2012 (C_{45w}, scaled for outages), finding that Hubble's .1 covers 2.2M

/24s, while our top-20 covers 2.5M, 14% more blocks. (We see similar results in observations from another site, and two months earlier, with C_{45c} and C_{44w}.)

6. EVALUATING INTERNET OUTAGES

We next apply our approach to measure Internet outages. We look at this data in two ways, first exploring event and outage durations then examining network wide stability by exploring marginal distributions ($\bar{\Omega}_B$ and $\bar{\Omega}_I$) across Internet space and time.

After correcting for local outages, we believe the observations in this section reflect Internet-wide stability, within the limits of measurement error. Since our vantage points are well connected and we remove local outages, our estimates approximate the Internet-core-to-edge reliability. We make this claim because we know our observations are stable across location and time (§5.5) and across all surveys in this section.

6.1 Evaluation over the Analyzable Internet

On 2011-09-28 we probed the entire analyzable Internet, targeting 20 samples in 2.5M blocks as described in §4.4. Somewhat surprisingly, this experiment drew no complaints, perhaps because it was shorter than our 2-week surveys. Data processing took 4 hours, both to visualize the results (as an image 2.5M × 134 pixels, broken into 20 tiles), and to detect the 946 routing events we observe. The overall outage rate is consistent with our survey data (§6.3): 0.3% outage area, or 99.7% availability. The absolute number of outages differ from Figure 7 roughly in proportion to different duration and scale. See Appendix E for a portion of the image for this dataset.

6.2 Durations and Sizes of Internet Outages and Events

We first consider the durations and sizes of block-level outages and network-wide events (Figure 9 left 2 plots).

Beginning with outages (Figure 9a), we see that half to three-quarters of outages last only a single round. Our current analysis limits precision to one round (11 minutes), but possible future work could examine individual probes to provide more precise timing. All surveys but Survey S_{39w} have the same trend; Survey S_{39w} diverges due to its local outages (dotted line S_{39w}), but joins the crowd when they are removed. We also see that 80% of outages last less than two hours. While there is no sharp knee in this distribution, we believe this time period is consistent with human timescales where operators detect and resolve problems.

Network events group individual outages by time, presumably due to a common cause. Figure 9b shows event durations, computed as the mean duration of each event's component outages. This figure shows that many single-round outages cluster into single-round events, since about 40% of events last one round instead of 50–75% of outages. With less strict clustering ($\theta = 10$ rounds instead of $\theta = 2$) this trend grows, with only 20% of events lasting one round.

About 60% of events are less than hour long, but there is a fairly long tail out to the limits of our observation (2 weeks or 20,000 minutes). This long tail is similar to distributions of event durations of Feamster et al. [17] and Hubble [23]. Feamster et al.'s very frequent probes (1–2 seconds between probes) in a mesh of computers, allow them to find 100% of events more than 10 s long, but the very high probing rate is only acceptable between a mesh of friendly computers. We cannot detect such short events, but we see the same long tail and our approach can scale to the whole Internet. Hubble favors large events, claiming to find 85% of events longer than 20 minutes and 95% of events longer than 1 hour. Our system captures all events longer than about 20 minutes (twice our probing interval), and about half of events from 10–20 minutes (Figure 6); more accurate than Hubble, particularly for shorter events.

Because local outages correspond to a single event, Survey S_{39w} resembles the other surveys both with and without removal of local outages, and Survey S_{39w} is indistinguishable from $S_{39w'}$.

Finally, we examined event sizes (figure in Appendix F). Almost all events are very small: 62% of events affect only a single block, and 95% are 4 blocks or smaller. Nevertheless, a few large outage events do occur, as discussed in §5.2.

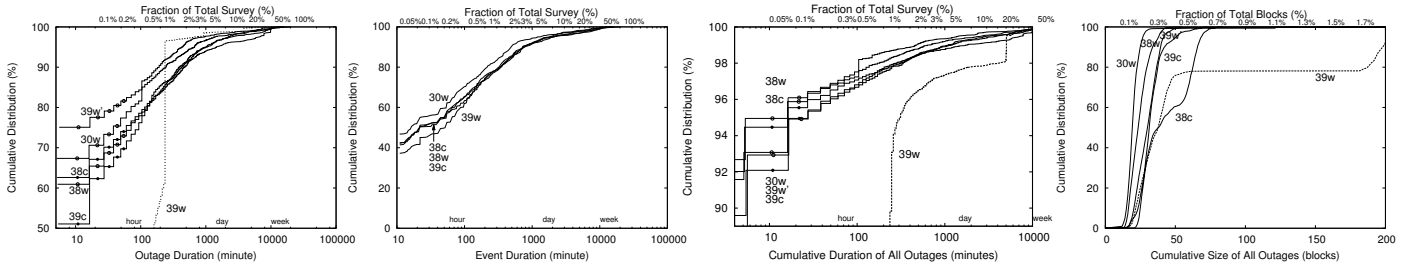
6.3 Internet-wide View of Outages

We next shift our attention to the Internet as a whole. How often is a typical block down, and how much of the Internet is inaccessible at any given time? To understand these questions, Figure 9 (right 2 plots) shows the marginal distributions of outages by round and block.

First we consider distribution by rounds in Figure 9c. As expected, we see the vast majority of the blocks in our survey are always up: from 92 to 95% of blocks have no outages over each two week observation. The exception is Survey S_{39w}, where two local outages partitioned the probers from the Internet for about two hours. When we remove local outages, this survey becomes consistent with the others. About 2% of blocks are out once (the step at 11 for one round) and the remaining tail follows the distribution of Figure 9c.

Turning to space, Figure 9d shows marginal distributions of $\bar{\Omega}_B$. Survey S_{39w} is again an outlier due to large local outages, but it resembles the others when local outages are removed.

Considering Figure 9d as a whole, we see that *almost always, some part of the Internet is inaccessible.* At any time, typically 20 to 40 blocks are unreachable in our



(a) Block outage durations. (b) Network event durations. (c) Marginal dist. by rounds, $\bar{\Omega}_I$. (d) Marginal dist. by blocks, $\bar{\Omega}_B$.

Figure 9: Cumulative distributions of outage and event durations (left two). Marginal distributions of outage, by round and block (right two). CDFs of (a) focus only on portions of the graph, CDFs of (c) starts at 80%. Datasets: surveys S_{30w}, S_{38c}, S_{38w}, S_{39c}, S_{39w}. The dotted lines are Survey S_{39w} without removing local outages.

survey. This result is consistent with our observations from Figure 7 that show 0.33% of the Internet is out, averaged over entire surveys, with a standard deviation of 0.1%. Our outage estimate is much lower than Paxson (up to 3.3% outages), suggesting much greater stability than 1995. It confirms the mesh study in RON [1] with a much larger number of edge networks. Finally, we see a set of unusually large outages in Survey S_{38c}, where the 50%ile outage is around 38 blocks, but 80%ile is at 63 blocks. We discuss the root causes for these outages in §5.2 and Figure 2.

Highly reliable networks are often evaluated in terms of availability, and compared to the "five nines" goal of telephone networks. We plot availability in the top panel of Figure 7, seeing that overall, the Internet is up about 99.7% of the time for about 2.5 nines of availability, suggesting some room for improvement.

The above analysis is based on surveys of 1% of the responsive Internet. We can confirm this result with our operational system scanning the entire analyzable Internet (§6.1), where we observed 0.3% of analyzable IPv4 space was out on average.

Our analysis of Internet-wide outages is preliminary, but it illustrates the utility of automated methods for detecting and quantifying outages in the data plane.

7. FUTURE WORK

Based on this work, a useful future direction is to study where in the network are outages located. Differentiating end-system outages (e.g., a company turning off its network) and close-to-core outages is important as the latter clearly has more impact.

Another interesting future direction is to look into diurnal and weekly patterns, in order to know how much of the Internet is "turned off" on a daily or weekly basis. Currently we regard such behaviors as "outages" by our definition. In the future, we can calculate the autocorrelations of the outage timeseries and decide if we see diurnal/weekly patterns.

8. CONCLUSIONS

Researchers have studied Internet outages with control- and data-plane observations for many years. We show that active probing of a sample of addresses in responsive /24 blocks provides a powerful new method to characterize network outages. We describe algorithms to visualize outages and cluster them into network-wide events. We validate this approach by both case studies and random samples, verify our results are stable and more accurate than prior work. With our system, a single PC can observe outages to destinations across the entire analyzable IPv4 Internet, providing a new approach to study Internet-wide reliability and typical outage size and duration.

Acknowledgments

We thank Jim Koda (ISI), Brian Yamaguchi (USC), and CSU network operations for providing BGP feeds to assist our evaluation, and Dan Massey, Christos Papadopoulos, Mikhail Strizhov for assisting with BGPmon and at CSU. We also thank Katsuhiro Horiba (WIDE) for providing probing infrastructure and BGP feeds. This work was reviewed by USC's IRB (IIR00000975) and identified as non-human subjects research.

9. REFERENCES

[1] David Andersen, Hari Balakrishnan, Frans Kaashoek, and Robert Morris. Resilient overlay networks. SOSP '01, pages 131–145.

[2] Grenville Armitage. Private communications, Jul. 2011.

[3] AusNOG. Discussions about australia flooding, Jan. 2011. `http://lists.ausnog.net/pipermail/ausnog/2011-January`.

[4] Randy Bush, James Hiebert, Olaf Maennel, Matthew Roughan, and Steve Uhlig. Testing the reachability of (new) address space. In *Proc. of ACM Workshop on Internet Nework Management*, August 2007.

[5] Randy Bush, Olaf Maennel, Matthew Roughan, and Steve Uhlig. Internet optometry: assessing

the broken glasses in internet reachability. In *Proc. of ACM IMC*, 2009.

[6] Xue Cai and John Heidemann. Understanding Block-level Address Usage in the Visible Internet. In *Proc. of SIGCOMM*, 2010.

[7] Di-Fa Chang, Ramesh Govindan, and John Heidemann. The Temporal and Topological Characteristics of BGP Path Changes. In *Proc. of ICNP*, November 2003.

[8] David R. Choffnes, Fabián E. Bustamante, and Zihui Ge. Crowdsourcing service-level network event monitoring. In *SIGCOMM*, 2010.

[9] James Cowie. Egypt leaves the Internet. `http://renesys.com/blog/2011/01/egypt-leaves-the-internet.shtml`, January 2011.

[10] James Cowie. Egypt returns to the Internet. `http://renesys.com/blog/2011/02/egypt-returns-to-the-internet.shtml`, February 2011.

[11] James Cowie. Libyan disconnect. Renesys Blog `http://renesys.com/blog/2011/02/libyan-disconnect-1.shtml`, February 2011.

[12] Ítalo Cunha, Renata Teixeira, Nick Feamster, and Christophe Diot. Measurement methods for fast and accurate blackhole identification with binary tomography. In *Proc. of 9thACM IMC*, 2009.

[13] A. Dainotti, R. Amman, E. Aben, and K. Claffy. Extracting benefit from harm: using malware pollution to analyze the impact of political and geophysical events on the Internet. *ACM Computer Communication Review*, Jan 2012.

[14] Alberto Dainotti, Claudio Squarcella, Emile Aben, Kimberly C. Claffy, Marco Chiesa, Michele Russo, and Antonio Pescapé. Analysis of country-wide internet outages caused by censorship. In *ACM IMC*, 2011.

[15] Amogh Dhamdhere, Renata Teixeira, Constantine Dovrolis, and Christophe Diot. NetDiagnoser: troubleshooting network unreachabilities using end-to-end probes and routing data. CoNEXT '07.

[16] Xun Fan and John Heidemann. Selecting Representative IP Addresses for Internet Topology Studies. In *ACM IMC*, 2010.

[17] Nick Feamster, David G. Andersen, Hari Balakrishnan, and Frans Kaashoek. Measuring the Effects of Internet Path Faults on Reactive Routing. In *ACM Sigmetrics - Performance*, 2003.

[18] Anja Feldmann, Olaf Maennel, Z. Morley Mao, Arthur Berger, and Bruce Maggs. Locating Internet Routing Instabilities. In *Proc. of SIGCOMM*, 2004.

[19] Ramesh Govindan and Hongsuda Tangmunarunkit. Heuristics for Internet Map Discovery. In *Proc. of IEEE Infocom*, March 2000.

[20] 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*, Oct. 2008.

[21] Yiyi Huang, Nick Feamster, Anukool Lakhina, and Jim (Jun) Xu. Diagnosing network disruptions with network-wide analysis. SIGMETRICS'07, pages 61–72.

[22] Yiyi Huang, Nick Feamster, and Renata Teixeira. Practical issues with using network tomography for fault diagnosis. *SIGCOMM Comput. Commun. Rev.*, 38:53–58, September 2008.

[23] Ethan Katz-Bassett, Harsha V. Madhyastha, John P. John, Arvind Krishnamurthy, David Wetherall, and Thomas Anderson. Studying black holes in the internet with Hubble. In *NSDI*, 2008.

[24] Ken Keys. Internet-scale IP alias resolution techniques. *ACM Computer Communication Review*, 40(1), January 2010.

[25] Ramana Rao Kompella, Jennifer Yates, Albert Greenberg, and Alex C. Snoeren. IP fault localization via risk modeling. In *NSDI*, 2005.

[26] Ramana Rao Kompella, Jennifer Yates, Albert Greenberg, and Alex C. Snoeren. Detection and Localization of Network Black Holes. In *Proc. of IEEE Infocom*, 2007.

[27] Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian. Delayed Internet routing convergence. In *Proc. of SIGCOMM*, 2000.

[28] Craig Labovitz, G. Robert Malan, and Farnam Jahanian. Internet routing instability. In *Proc. of SIGCOMM*, 1997.

[29] Derek Leonard and Dmitri Loguinov. Demystifying service discovery: Implementing an internet-wide scanner. In *ACM IMC*, 2010.

[30] Harsha V. Madhyastha, Tomas Isdal, Michael Piatek, Colin Dixon, Thomas Anderson, Arvind Krishnamurthy, and Arun Venkataramani. iPlane: an information plane for distributed services. In *OSDI*, 2006.

[31] Ratul Mahajan, David Wetherall, and Tom Anderson. Understanding BGP misconfiguration. In *Proc. of SIGCOMM*, 2002.

[32] Om Malik. In Japan, many undersea cables are damaged. GigaOM blog, `http://gigaom.com/broadband/in-japan-many-under-sea-cables-are-damaged/`, Mar. 14 2011.

[33] Athina Markopoulou, Gianluca Iannaccone, Supratik Bhattacharyya, Chen nee Chuah, and Christophe Diot. Characterization of Failures in an IP Backbone. In *Proc. of IEEE Infocom*, 2004.

[34] University of Oregon. Route Views Project. `http://routeviews.org/`.

[35] Vern Paxson. End-to-end routing behavior in the internet. SIGCOMM'96, pages 25–38.

[36] Lin Quan and John Heidemann. Detecting internet outages with active probing. Technical Report ISI-TR-2011-672, May 2011.
[37] Yuval Shavitt and Eran Shir. DIMES: let the internet measure itself. *ACM Computer Communication Review*, 35, Oct. 2005.
[38] Renata Teixeira and Jennifer Rexford. A measurement framework for pin-pointing routing changes. In *Proc. of the ACM SIGCOMM workshop on Network troubleshooting*, 2004.
[39] International Business Times. Optus, Telstra see service outages after Cyclone Yasi, 2011. `http://hken.ibtimes.com/articles/108249/20110203/optus-telstra-see-service-outages-after-cyclone-yasi.htm`.
[40] Los Angeles Times. Amazon apologizes for temporary server outage. `http://www.latimes.com/business/la-fi-amazon-apology-20110430,0,4604776.story`.
[41] New York Times. Egypt cuts off most internet and cell service. `http://www.nytimes.com/2011/01/29/technology/internet/29cutoff.html`.
[42] Daniel Turner, Kirill Levchenko, Alex C. Snoeren, and Stefan Savage. California fault lines: understanding the causes and impact of network failures. In *Proc. of SIGCOMM*, 2010.
[43] USC/LANDER project. Internet address survey dataset, predict id `usc-lander/internet_address_survey_reprobing`, April 2011.
[44] Webnet77. IpToCountry database, March 2011. `http://software77.net/geo-ip/`.
[45] Eric Wustrow, Manish Karir, Michael Bailey, Farnam Jahanian, and Geoff Huston. Internet background radiation revisited. In *ACM IMC*, 2010.
[46] He Yan, Ricardo Oliveira, Kevin Burnett, Dave Matthews, Lixia Zhang, and Dan Massey. BGPmon: A real-time, scalable, extensible monitoring system. In *Proc. of IEEE CATCH*, March 2009.
[47] Ming Zhang, Chi Zhang, Vivek Pai, Larry Peterson, and Randy Wang. PlanetSeer: Internet Path Failure Monitoring and Characterization in Wide-area Services. In *OSDI*, 2004.

APPENDIX

These appendices contain secondary material supporting the main paper.

A. FULL LIST OF DATASETS

Table 4 lists all the datasets we study, and what fraction of each dataset is analyzable.

All datasets are available at no cost from the authors and through the PREDICT program, `http://www.predict.org`. In PREDICT, each dataset has PRE-

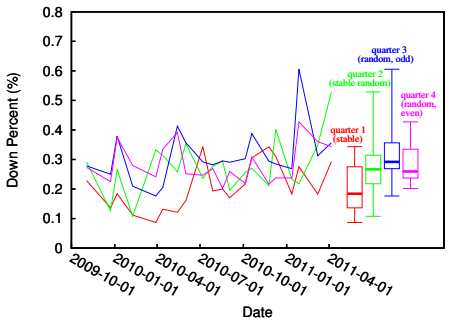


Figure 10: Downtime percentage over time, for 4 different quarters of our dataset, from S_{29w} to S_{40w}.

DICT id: `PREDICT/USC-LANDER/internet_address_survey_reprobing_it29w-20091102`, or equivalent for different survey numbers and dates.

B. EFFECT OF BLOCK SELECTION METHOD

In §3.1 we discuss the selection methodology for surveys. A full description was presented in 2006 [20]. While three-quarters of blocks are randomly selected, one quarter are chosen to represent a range of conditions. In §5.1 we state that selection of this quarter results in a slight underestimate of outages. We next present the analysis to quantify that statement.

Target blocks in each survey can be grouped into four quarters: stable and selected to represent different characteristics; stable but randomly selected; randomly chosen each survey, with an odd third octet; and randomly chosen each survey, with an even third octet. To validate if our results are skewed by selection of blocks, we plot the outage percentage of the four quarters over time (Figure 10). We also plot the outage percentage quartiles of all four quarters in the right part of Figure 10 (for raw data, please see Table 5), showing we are slightly under-estimating the Internet's outages, as Quarter 1 (stable fixed blocks) has less overall outage rates (0.2%, with standard deviation 0.078%), while other three quarters' outage rates are around 0.29% (standard deviation 0.08%).

While this comparison shows a slight bias for the stable-selected blocks, this bias is slight and affects only one quarter of all observed blocks, so our overall conclusions are only slightly more stable than a random sample would be. Overall outage estimates from surveys appear slightly more stable (about 0.06% less downtime) than would analysis of a completely random sample.

C. PROBING SYSTEM PERFORMANCE

To show our system can probe the entire analyzable Internet, we evaluated raw prober performance. For this experiment we use a 4-core Opteron with 8GB memory system to probe a set of IP addresses rang-

Survey	Start Date	Duration (days)	Blocks (Analyzable)
S_{29w}	2009-11-02	14	22371 (46%)
S_{29c}	2009-11-17	14	22371 (45%)
S_{30w}	2009-12-23	14	22381 (47%)
S_{30c}	2010-01-06	14	22381 (48%)
S_{31w}	2010-02-08	14	22376 (48%)
S_{31c}	2010-02-26	14	22376 (49%)
S_{32w}	2010-03-29	14	22377 (48%)
S_{32c}	2010-04-13	14	22377 (48%)
S_{33w}	2010-05-14	14	22377 (48%)
S_{33c}	2010-06-01	14	22377 (48%)
S_{34w}	2010-07-07	14	22376 (47%)
S_{34c}	2010-07-28	14	22376 (47%)
S_{35w}	2010-08-18	14	22376 (47%)
S_{35c}	2010-09-02	14	22375 (47%)
S_{36w}	2010-10-05	14	22375 (48%)
S_{36c}	2010-10-19	14	22375 (48%)
S_{37w}	2010-11-24	14	22374 (48%)
S_{37c}	2010-12-09	14	22373 (48%)
S_{38w}	2011-01-12	14	22375 (47%)
S_{38c}	2011-01-27	14	22373 (47%)
S_{39w}	2011-02-20	16	22375 (52%)
S_{39c}	2011-03-08	14	22375 (49%)
S_{39w2}	2011-03-22	14	22374 (49%)
S_{40w}	2011-04-06	14	22922 (47%)
S_{40c}	2011-04-20	14	22921 (47%)
S_{41w}	2011-05-20	14	40645 (57%)
S_{41c}	2011-06-06	14	40639 (57%)
S_{42w}	2011-07-26	14	40565 (52%)
S_{42c}	2011-08-09	14	40566 (56%)
S_{43w}	2011-09-13	14	40598 (53%)
S_{43c}	2011-09-27	14	40597 (56%)
$S_{AnalyzableInternet}$	2011-09-28	1	2.5M (100%)
S_{43j}	2011-10-12	14	40594 (54%)
S_{44w}	2011-11-02	14	40634 (57%)
S_{44c}	2011-11-16	14	40632 (57%)
S_{44j}	2011-12-05	14	40631 (56%)

Table 4: Internet surveys used in this paper, with dates and durations. Survey numbers are sequential with a letter indicating collection location (w: ISI-west in Marina del Rey, CA; c: Colorado State U. in Ft. Collins, CO; j: Keio University, Tokyo, Japan). Blocks are analyzable if $\bar{C} \geq 0.1$.

Quarter	Mean	Min	Max	q1	q2	q3
1 (stable)	0.21	0.09	0.34	0.14	0.18	0.28
2 (stable but random)	0.28	0.11	0.53	0.22	0.27	0.31
3 (random, odd third octet)	0.31	0.18	0.61	0.27	0.29	0.36
4 (random, even third octet)	0.29	0.20	0.43	0.24	0.26	0.33

Table 5: Outage percentage statistics of four quarters, from S_{29w} to S_{40w}.

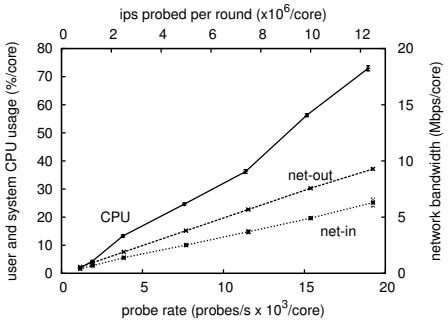


Figure 11: Performance of one prober instance as number of targets grows: 1-core CPU (left scale) and bandwidth (right).

ing in number from 1M to about 50M, taken from our optimized set of sampled addresses and target blocks.

Assuming a good Internet connection, we are primarily CPU constrained, as the prober manages data structures to match responses with requests to confirm the probed addresses.

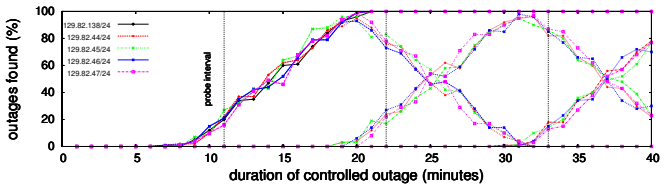
Figure 11 shows single-core CPU load and network traffic for one instance of our prober as we increase the number of target addresses per round. Each observation shows the mean and a very small standard deviation from 18 measurements taken every minute, starting 13 minutes into a probing run to avoid startup transients. Memory is fixed at roughly 333MB/core, growing linearly from 325MB to 346MB over this range of probe rates.

Fortunately, probing parallelizes easily; in operation we run four parallel probers: each a separate process (on a different CPU core), probing a separate part of address space. There is minimal interference between concurrent jobs, and in fact the data from Figure 11 reflects 4-way parallelism. Our 4-way probing therefore meets our target of 75k probes/s to cover the sampled Internet at $k = 20$ per block.

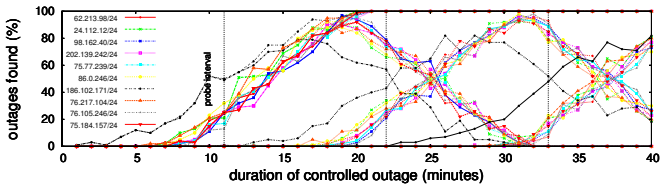
D. CONTROLLED OUTAGES FOR OTHER BLOCKS

In §5.4 we verified outage detection through controlled experiments. To confirm that the block we report there is representative, we selected four additional block at CSU, and with ten randomly chosen blocks from around the Internet. As before, in our datasets each of these blocks is evaluated as always available across our observation period, although each has a different number of responsive hosts and random packet loss. We then repeat our experiment, artificially injecting outages and evaluating what our algorithms observe.

Figure 12a shows the result of controlled outages in all five CSU blocks. To make it easier to compare different blocks, we connect the percentage of estimates for each block with a line rather than plotting 5 separate bars.



(a) Comparing results of emulated outages for 5 CSU blocks.



(b) Comparing results of emulated outages for 10 random blocks.

Figure 12: Comparing results of emulated outages for 5 CSU blocks (top) and 10 random blocks (bottom).

We see that the trends in outage detection are within a few percent across all blocks, suggesting that the results shown in Figure 6 are representative.

To further validate if our results are stable, we randomly picked 10 /24 blocks that were judged always up, and we do the same controlled outage experiment. Figure 12b shows this experiment. Here almost all blocks show similar results as Figure 12a. One block, 186.102.171/24, has lower outage estimates than the others. Based on examination of a 2-week survey, we believe this block uses dynamically assigned addresses, only about 15% of which are occupied. Therefore we see few responses in our sample (typically only 3 of 20), and variation as addresses are reassigned affect our conclusions. Improving our results for dynamically assigned blocks is ongoing work. We conclude that for responsive blocks are results are quite consistent, while variation in our estimates is greater in sparse and dynamic blocks.

E. SELECTED PORTIONS OF OUTAGES IN THE ENTIRE ANALYZABLE INTERNET

On 2011-09-28, we probed the entire analyzable Internet for 24 hours, targeting 20 samples in 2.5M blocks (described in §4.4 and §6.1). Figure 13 shows selected portions of outages in this survey, as they are well correlated and affect many blocks. (We omit most of the plot; a complete plot at 600 dots-per-inch would be more than 375 pages long.)

Figure 13a shows an outage in a Brazilian AS (AS26615) from 2011-09-02 T03:34 +0000 for 25 rounds (about 4.5 hours), affecting more than 350 /24 blocks. We are able to partially verify this outage with BGP control plane messages.

The other three parts of this figure show outages affecting more than 800 /24 blocks in southern China (Figures 13b and 13c), including 35 /24 blocks in a

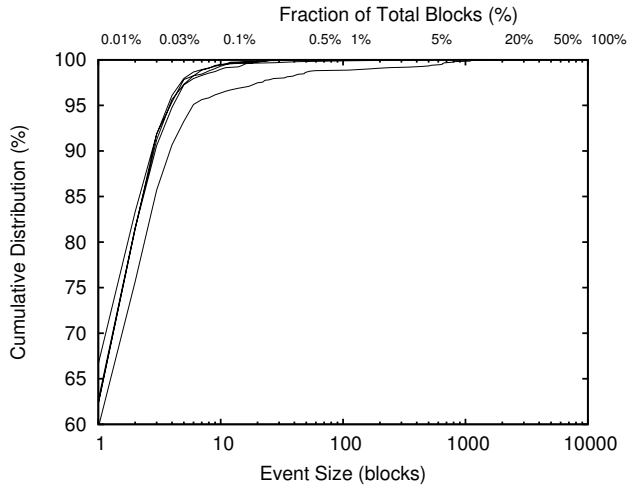
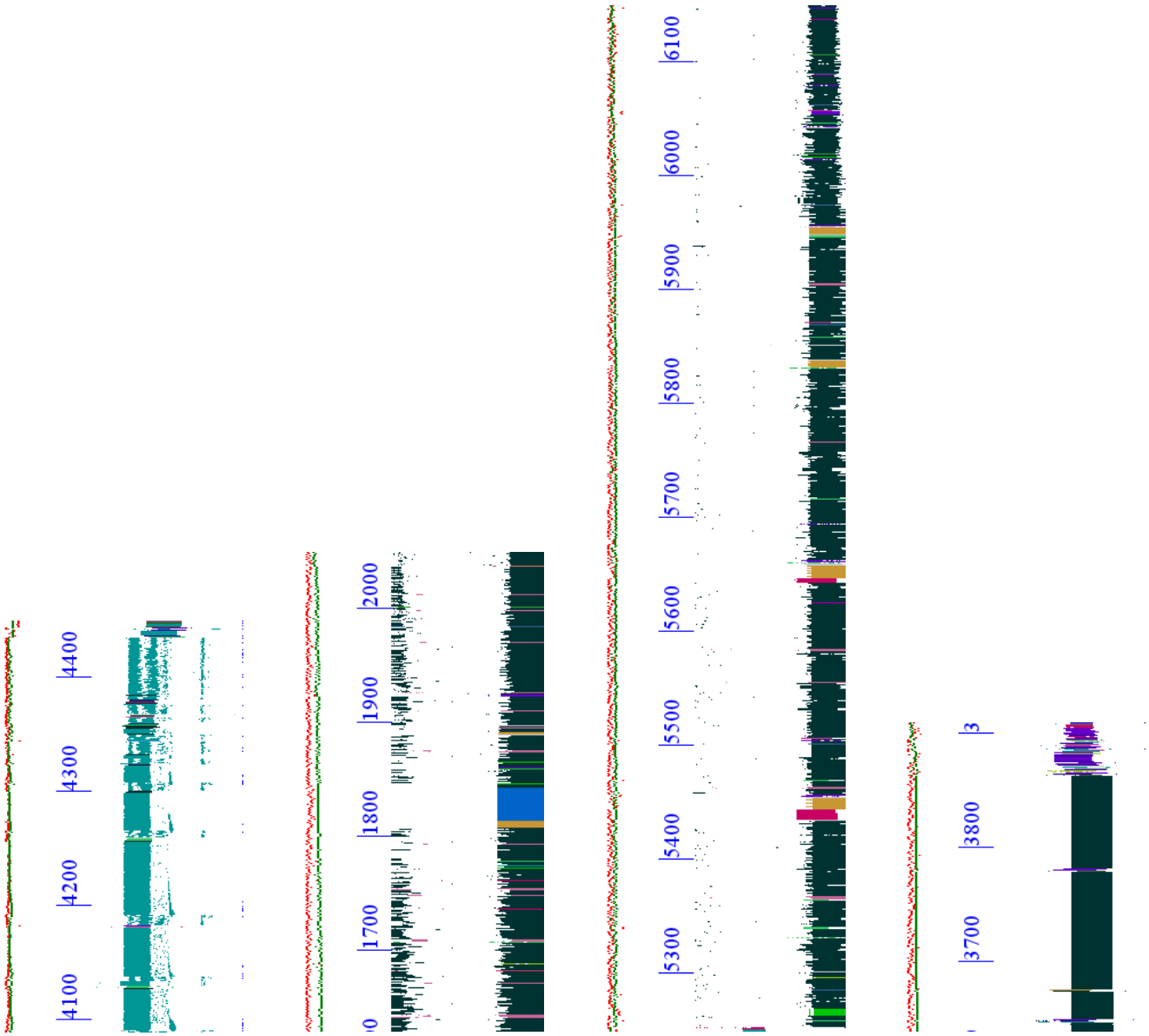


Figure 14: Network event sizes, cumulative distributions of outage and event durations, from Surveys S_{30w}, S_{38c}, S_{38w}, S_{39c}, S_{39w}.

mass-transit Internet (as part of Figure 13d). We did not observe evidence for these outages in BGP, but did correlate their timing and location with news reports confirmed in international media.

F. NETWORK EVENT SIZES

Extending Figure 9, discussed in Section 6.2, Figure 14 shows the distribution of network event sizes.



(a) Portion 1 (b) Portion 2 (c) Portion 3 (d) Portion 4

Figure 13: Selected slices of outages in the analyzable Internet study. Colored regions show 4.5–7.3 hours (25–45 rounds) of the 24 hours measurement (133 rounds). Each of the X axis is 24 hours in time. Subgraphs on the Y axis show marginal distribution (green line) and overall block responsiveness (red dots).