Less is More: Optimizing Probe Selection Using Shared Latency Anomalies

TAVEESH SHARMA, University of Chicago, USA
ANDREW CHU, University of Chicago, USA
PAUL SCHMITT, Cal Poly, USA
FRANCESCO BRONZINO, ENS Lyon / Institut universitaire de France, France
NICOLE P. MARWELL, University of Chicago, USA
NICK FEAMSTER, University of Chicago, USA

Latency anomalies—persistent or transient increases in round-trip time (RTT)—are a common feature of residential Internet performance. When multiple users simultaneously experience anomalies at the same destination, it may indicate shared infrastructure issues, routing behavior, or congestion. However, inferring such shared behavior is challenging in practice. This is because the magnitude of these anomalies can vary significantly across devices, even within the same ISP and geographic area, and detailed network topology information is often unavailable due to platform limitations or privacy constraints.

In this work, we study whether devices that experience a shared latency anomaly observe similar changes in RTT magnitude using a topology-agnostic approach. Using a four-month dataset of high-frequency RTT measurements from 99 residential probes in Chicago, we detect shared anomalies and analyze their consistency in amplitude and duration without relying on traceroutes or explicit path information. Building on prior change-point detection techniques, we find that many shared anomalies affect users similarly in amplitude, particularly within the same ISP. Leveraging this insight, we develop a sampling algorithm that reduces redundancy in detected anomalies by selecting representative devices under user-defined constraints. Our approach covers 95% of aggregate anomaly impact with less than half the total probes used in our deployment. Compared to two baselines, we show that our approach selects a significantly higher number of unique anomalies at similar coverage levels. Additionally, our analysis suggests that geographic diversity can play an important role in selecting probes for a single ISP even within a single city. These findings highlight the potential of using anomaly amplitude and duration as topology-independent signals for scalable monitoring, troubleshooting, and cost-effective sampling designs in residential Internet performance measurement.

CCS Concepts: • **Networks** → **Network performance analysis**; **Network measurement**.

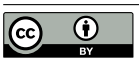
Additional Key Words and Phrases: Change Point Detection, Latency, Network Observability, Probe Selection

ACM Reference Format:
Taveesh Sharma, Andrew Chu, Paul Schmitt, Francesco Bronzino, Nicole P. Marwell, and Nick Feamster. 2026. Less is More: Optimizing Probe Selection Using Shared Latency Anomalies. *Proc. ACM Netw.* 4, CoNEXT2, Article 18 (June 2026), 25 pages. https://doi.org/10.1145/3808666

1 Introduction

Residential broadband network performance has been an area of active research and policy interest [47, 55, 64, 71, 86, 88], given its critical role in enabling access to several aspects of digital life. While

Authors' Contact Information: Taveesh Sharma, University of Chicago, Chicago, Illinois, USA; Andrew Chu, University of Chicago, Chicago, Illinois, USA; Paul Schmitt, Cal Poly, San Luis Obispo, California, USA; Francesco Bronzino, ENS Lyon / Institut universitaire de France, Lyon, France; Nicole P. Marwell, University of Chicago, Chicago, Illinois, USA; Nick Feamster, University of Chicago, Chicago, Illinois, USA.



ACM 2834-5509/2026/6-ART18
https://doi.org/10.1145/3808666

aggregate Internet performance has improved over the years [19, 60], Internet users continue to experience frequent performance disruptions in the form of low speed, elevated latency, and packet loss [21, 30, 55]. Among these, latency anomalies—transient as well as persistent increases in round-trip time (RTT)—particularly affect latency-sensitive applications such as video conferencing [53, 84], online gaming [3], and other real-time services.

Latency anomalies can significantly degrade user experience without causing complete service outages. Because they do not operate as binary reachability failures, such anomalies are often invisible to coarse-grained availability metrics (e.g., loss of connectivity, prefix withdrawals, or host reachability) commonly used by traditional outage detection systems [77, 92]. As a result, operators and policymakers may underestimate both the prevalence and the impact of performance degradations that affect users without triggering alerts.

Prior research suggests that latency anomalies can arise from routing changes [15], server-side anomalies [81], shared infrastructure bottlenecks [85], congestion [24], or other network phenomena. Robust techniques have been developed to detect such events using statistical change-point detection applied to longitudinal measurements [15, 24, 52]. These studies analyze anomalies at the level of individual probe-destination pairs or rely on explicit path or topology information (e.g., traceroutes) to reason about shared infrastructure. Unfortunately, in many practical residential network measurement deployments, traceroutes are unavailable due to privacy constraints, platform limitations, or measurement overhead, leaving operators with little visibility into shared paths. For instance, Measurement Lab (M-Lab) [61] collects server-to-client traceroutes for speed tests opportunistically, and are not guaranteed to be available for all measurements. Further, Ookla's Open Data [67] provides only aggregated client-side measurements without any path information. Finally, while RIPE Atlas [80] provides access to some traceroute data, it is difficult to find cities with a sufficiently large number of probes to conduct a meaningful within-city analysis. In such settings, it remains unclear how to identify redundancy among probes or select representative devices in a principled manner.

In this paper, we pose a largely unexplored question: *when two or more residential devices experience a latency anomaly to the same destination at the same time, do they experience it with similar magnitude?* Unlike an array of prior work that characterizes network performance at coarse granularities (city- [68], metro- [14], AS/ISP- [22], or country-level [16]), our focus on inter-probe correlation within a city enables a microscopic understanding of how the digital divide manifests in residential broadband networks. While the digital divide is often framed in terms of access, affordability, and adoption, recent work [12, 29, 71] has increasingly emphasized that the quality and reliability of connectivity are also important dimensions of meaningful Internet access. Further, we ask this question without assuming access to traceroutes or explicit topology information. Instead, we rely solely on end-to-end latency measurements and their spatio-temporal structure to infer shared behavior across probes from a dense urban deployment. This design reflects the constraints faced by many operational deployments and allows us to study probe redundancy in a topology-free manner. This question is important for two reasons. First, if latency anomalies are spatially correlated and exhibit similar amplitude across devices, then analyzing every probe independently introduces substantial redundancy. Second, understanding the consistency of anomaly magnitude across devices can provide valuable diagnostic signals for localizing faults (e.g., last-mile versus middle-mile bottlenecks), even in the absence of path-level visibility. If such consistency exists, measurements from a small number of representative probes may suffice to characterize the broader impact of network events.

To explore these questions, we analyze a four-month dataset of high-frequency RTT measurements collected from 99 fixed residential probes deployed in home networks across Chicago. Each probe measures RTT to a common set of destinations at five-minute intervals. We primarily focus on latency to M-Lab servers [61], whose static IP addresses and global footprint allow us to detect network-induced anomalies while minimizing noise from DNS resolution or server-side dynamics.

Building on prior work [15, 24], we apply change-point detection techniques to identify latency anomalies and analyze the relationship between their temporal overlap and similarity in amplitude across probes.

We then formulate the problem of probe selection as a maximum weighted set coverage problem [44, 65], where sets correspond to distinct latency anomalies and weights capture their impact. Unlike prior approaches that rely on inferred paths or shared links, our formulation operates entirely on anomaly co-occurrence and impact. It requires no knowledge of the underlying network topology. To approximate the optimal solution to this NP-hard problem, we leverage a greedy heuristic that maximizes coverage of distinct anomalies while minimizing the number of selected devices.

Our results show that shared anomalies often exhibit similar amplitudes across devices, particularly when users share the same ISP, and that anomalies with greater temporal overlap tend to have higher impact. These findings work towards answering a practical question faced by operators and researchers alike: *can we reduce measurement redundancy in existing deployments without losing visibility into important performance degradations?* Reducing redundancy lowers data collection and processing costs, simplifies downstream analyzes, and mitigates noise introduced by multiple probes observing the same event. In this sense, probe selection acts as a form of spatio-temporal down-sampling that preserves the most operationally relevant signals in large-scale measurement datasets.

To quantify these trade-offs, we evaluate how many devices are needed to capture the majority of observed anomaly impact, which we define as the product of amplitude and duration. We find that capturing 95% of total impact requires fewer than half of the probes in our deployment (44 out of 99). Compared to uniform random selection, which selects 33 probes, our approach detects 2.2× more unique anomalies toward a local M-Lab server. We further show that a small amount of historical data (1–2 weeks) is sufficient to select probes that continue to provide steady anomaly coverage in the future. Assuming that the overhead of computing shared anomalies and selecting probes is low for a deployment as ours, these results indicate that careful, topology-agnostic probe selection can significantly reduce measurement cost. At the same time, it preserves the ability to detect and characterize impactful latency anomalies in residential broadband networks. This is especially important for monitoring infrastructure deployed by network operators in cities, which often consists of several probes that are expensive to maintain and operate.

In our knowledge, this is the first work that studies co-occurrence of latency anomalies across a citywide deployment to inform probe selection without relying on path or topology information. We make the following contributions:

- We show that latency anomalies observed by different residential probes often exhibit similar amplitudes and durations when they temporally overlap, even without access to path or topology information (Section 4.3).
- We develop a topology-agnostic probe selection algorithm that leverages temporal overlap and anomaly impact to identify representative devices in longitudinal measurement deployments (Section 5.2).
- We demonstrate that our algorithm covers 95% of anomaly impact using fewer than half of the probes in our deployment and significantly outperforms baseline selection strategies in detecting unique anomalies (Section 5.3).
- We show that a small amount of historical data (1–2 weeks) is sufficient to select probes that continue to provide steady anomaly coverage in the future (Section 6).

2 Background & Related Work

This section surveys prior work that informs our methodology and contributions. We begin by reviewing studies on broadband Internet performance and Internet topology mapping. We then

describe change-point detection techniques for latency time-series data and detail the foundation of our approach, including limitations of prior work that motivate our modifications. Next, we discuss strategies for probe selection in large-scale active measurement platforms, emphasizing how our data-driven approach differs from topology-based methods. Finally, we outline the maximum weighted set coverage problem, a classical optimization problem that underpins our formulation for selecting representative probes.

2.1 Broadband Internet Performance

Several prior studies have analyzed broadband Internet performance using large-scale measurement platforms [14, 18, 89]. The FCC's Measuring Broadband America (FCC MBA) program [27] and M-Lab [61] provide insight into ISP-level speed and latency characteristics for individual connections, including variation by geography and over time. Sundaresan *et al.* [88] show that broadband performance shows significant deviation from advertised speeds, particularly during peak hours. More recent research has shifted the focus from understanding a single end-to-end path to analyzing Internet performance dynamics for collective user populations. For instance, CableMon [36] groups cable modems connected to the same fiber node that share similar proactive network maintenance (PNM) metrics. TelApart [35] focuses on differentiating between performance issues that are specific to a single user and those that affect multiple users in the same area. While our work shares the same motivation of understanding groups of users with similar performance issues, our focus is on leveraging similarities in end-to-end latency to reduce monitoring redundancy. Other similar studies can broadly be classified into regional performance comparisons [72, 83], mapping efforts [12, 82], statistical modeling [38, 47, 86], and policy focused work [21, 54, 55, 57, 58]. While these studies recognize the importance of understanding the spatio-temporal dynamics of Internet performance, our work is the first to utilize shared latency anomalies to reduce redundancy in measurement infrastructure.

2.2 Internet Topology Mapping

A number of efforts have been made to better understand Internet topology, both in regard to its structure (connectivity-based and geographic) [17, 32] and state [8, 25, 31, 40, 41, 78, 79, 93]. In this space exist a number of high-level parallels to our objectives. One such objective is improving measurement scalability while retaining fidelity and resilience against false positives. Hu *et al.* [37] present an algorithm for geolocating IP addresses that selects the best few vantage points to measure from while maintaining accuracy comparable to approaches that rely on more measurements. This effort is very similar to our work, with vantage points being tantamount to the residential probe devices described in this paper. Our work differs in that we attempt to efficiently and accurately measure latency anomalies as compared to exploring how to scale up IP geolocation algorithms. Various other works also examine this problem in the context of active probing [69, 75, 76].

Another parallel objective is determining if similarities exist between endpoints that share spatio-temporal characteristics, and the implications of these similarities. Cai *et al.* [13] develop a network address block clustering technique which reveals that contiguous addresses share qualities such as utilization and link speed. Similarly, Baltra *et al.* [9] develop two algorithms for detecting groups of IP addresses with differing connectivity characteristics (islands and peninsulas), and use these taxonomies to improve the DNSMon tool [6]. Our efforts in this paper hope to answer a similar question, but in regard to residential Internet performance—do devices that experience the same latency anomaly experience it at the same magnitude? We detail our findings and their implications towards answering this question in Section 5.3.

2.3 Latency Time-series Change-point Detection

Detecting change-points in latency time-series is a widely used technique for several applications. These include detecting congestion and path changes [15, 24, 34, 52], detecting routing attacks [23, 39], adaptive network management [51], and application performance monitoring [20, 28]. A variety of statistical and algorithmic techniques have been proposed, ranging from CUSUM [90] and Bayesian change-point models [91] to signal decomposition and machine learning based approaches [11, 45].

Our methodology for detecting latency anomalies in an RTT time-series is an extension of Jitterbug [15]. Jitterbug was originally designed to detect changes in RTT for inter-domain links. It first uses Bayesian change-point detection (BCP) [91] or Hidden Markov Models (HMMs) [63] to identify a set of candidate timestamps where a change in latency occurs. Here, a change encompasses both positive and negative shifts in mean latency. Then, it uses a set of heuristics to filter positive changes that are less likely to be caused by a significant network event. The heuristics include inferring a “jump" in latency by checking if the difference in means of consecutive segments is greater than a threshold (the recommendation is 0.5 ms), and checking whether dispersion in jitter is greater than a threshold to signal congestion. At the time of writing, Jitterbug updated its implementation to add support for a range of faster algorithms for change-point detection, including Pruned Exact Linear Time (PELT) [45], Rbeast [94], and rule-based methods such as ADTK [5]. We benchmark our approach against some of these variants in Section 4.2.

While we acknowledge the effectiveness of Jitterbug in distinguishing congestion events from other network phenomena, we make a number of modifications to make this approach more suitable for our use case. First, as the authors note, Jitterbug’s range of supported change-point detection methods do not detect *all* periods of elevated latency. In our work, we make an attempt to address this limitation by introducing new heuristics and detection methods with greater sensitivity to flag more jumps. Second, we notice in our experiments that Jitterbug applies change-point detection on the entire time-series at once, which may result false-negatives. For example, if change-points are located at the edges of the RTT time-series, they are likely to be considered as “normal" behavior due to limited sample availability around boundaries. Since we are interested in identifying shared network anomalies across devices, it is important to ensure that we detect maximum changes in latency. Finally, Jitterbug was originally designed to infer network congestion in access networks. Our work, on the other hand, is focused on probe selection after identifying similarities in *all* latency anomalies across devices, which may not necessarily be caused by congestion. We detail these modifications in Section 4.1.

2.4 Probe Selection

Selection of representative probes in active Internet measurement has been extensively studied, with strategies ranging from improving geographic footprint to more sophisticated methods that also take coverage into account. Akella and Seshan [2] show that using only academic testbeds like PlanetLab can bias results, and propose selecting probes based on traffic patterns and popular destinations. Barford *et al.* [10] find that adding more probes has diminishing returns, suggesting that a small set of well-placed probes can provide autonomous system (AS)-level visibility. Recent work focuses on maximizing coverage and diversity: Holterbach *et al.* [33] proposed selecting topologically dissimilar probes, and the Metis algorithm improves RIPE Atlas coverage by targeting underrepresented ASes and regions [4, 7]. Others use BGP or topology data to choose the minimal set of probes that observe the most paths [87]. Our work differs from these studies by relying on no prior knowledge of the network topology. We use purely an end-to-end, data-driven approach to identify a minimal set of probes that can cover shared latency anomalies.

2.5 Maximum Weighted Set Coverage Problem

The maximum weighted set coverage problem is a widely studied combinatorial optimization problem [44, 65]. Given a set of items U and a collection of subsets $S_1, S_2, \ldots, S_n$ such that $S_i \subseteq U$, the goal is to select a subset of these sets such that the union of the selected sets covers as many items in U as possible. Each set S_i has an associated weight w_i, and the objective is to maximize the total weight of the selected sets while ensuring that each item in U is covered at least once.

An important property of this problem is that it is NP-hard, implying that there is no known polynomial-time algorithm that guarantees an optimal solution. However, there are several approximation algorithms that provide close-to-optimal solutions in polynomial time. One of the most common approaches is the greedy approach, which iteratively selects the set that covers the largest additional weight of uncovered elements at each step. This approach continues until the desired number of sets (up to a given budget k) is selected. Nemhauser et al. [65] showed that this greedy algorithm achieves a $(1 - 1/e) \approx 63\%$ approximation to the optimal solution, which is the best possible guarantee unless $P = NP$. In Section 5.2, we propose a similar greedy algorithm for selecting representative probes in our deployment.

This problem has applications in various fields, including outbreak detection [49], sensor placement [46], social network analysis [43], and text summarization [50]. In our work, we reformulate the problem of selecting representative devices as an instance of this problem, where the items in each set are latency anomalies, which may or may not be shared across devices. The weights of the sets are determined by an "impact" metric, which is a function of the amplitude and duration of the shared anomalies. The goal is to select a subset of devices that maximizes the total impact while minimizing the number of unique anomalies detected in the selected devices. We posit that the definition of impact, albeit simple in our case, can be extended to more nuanced definitions in future work. For example, one could consider defining impact in terms of QoE degradations [56, 62, 84] for a particular application, or in terms of the geographic area of occurrence for a particular latency anomaly. We omit these explorations in our current work, but we believe they are important avenues for future research.

3 Dataset

In this section, we outline our data collection approach and share basic descriptive statistics of our dataset. We conclude with a discussion on the capability of our dataset to capture shared network anomalies.

3.1 Data Collection and Measurement Frequency

We collect the data for this study primarily from residential networks using an open-source measurement platform[1]. The platform allows for the collection of a wide suite of network performance tests such as throughput, latency, application Quality of Experience (QoE) and network path metrics. Our platform is typically deployed on a Raspberry Pi 4B (RPi) probe that connects directly to the home router using an Ethernet cable. The probe runs a set of tests periodically to measure various network performance metrics. While RPis are resource-constrained devices, prior benchmarking work [70] has shown the RPi 4B to achieve gigabit speeds on wired connections, which is suitable for running network performance tests. Further, our focus in this work is on end-to-end `ping` latency measurements, which are not significantly affected by the RPi's performance. We also note that our measurement suite can be easily adapted to run on other platforms.

We leverage latency measurements from our platform to identify the presence of shared network anomalies among 99 user devices deployed between April 2022 and July 2022. Although our devices

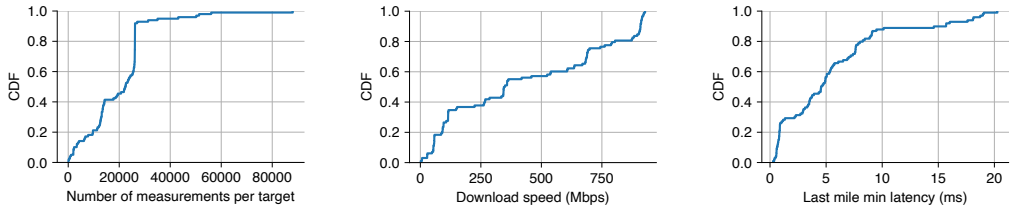
[1]We withhold the platform name to preserve our submission's anonymity.

Destination	IP Address
Atlanta	4.71.254.129
Chicago	4.71.251.129
Denver	4.34.58.1
Johannesburg	196.24.45.129
Paris	77.67.119.129
Seattle	38.102.0.65
Stockholm	195.89.146.193
Tunis	41.231.21.1

Table 1. Destinations used for latency measurements.

ISP	Number of devices
Comcast	62
AT&T	27
RCN	4
Everywhere Wireless	2
Webpass	2
Campus Network	1
Verizon	1

Table 2. Distribution of the number of devices by ISP.



(a) Measurement count per probe for last-mile latency.

(b) Median download speed distribution per probe.

(c) Minimum last-mile latency distribution per probe.

Fig. 1. An overview of basic descriptives of our dataset of 99 devices.

are deployed even during the present day, we use this time period because it is when most of our devices were actively collecting data. Each probe reports a new ICMP round-trip latency measurement every five minutes, and performs one NDT and one Ookla speed test every hour. We primarily use latency measurements destined to M-Lab servers as these provide a global coverage along with being destinations with fixed IP addresses. Table 1 shows the list of destinations used for latency measurements that we analyze for this work. The destinations are chosen to be geographically diverse, with a mix of locations in North America, Europe, and Africa, to guarantee topological diversity. Each selected server is located outside the AS of the probe's ISP to ensure that we are measuring the end-to-end path and not just the last-mile or middle-mile.

3.2 Dataset Overview

Our deployment consists of 99 Raspberry Pi devices located in the city of Chicago. These are distributed across 31 distinct zip codes, with the majority concentrated in just two. We analyze devices at the zip code level because this is the finest spatial resolution commonly available in IP geolocation databases beyond the city level. As a result, our approach remains applicable to studies that rely on coarse location metadata rather than precise probe coordinates. The remaining zip codes each contain between one and six devices, and we use data from these devices to validate our findings. A well-documented contrast in terms of social demographics between the two zip codes motivated this sampling decision [42]. Despite this heterogeneity, we do not constrain our methods to look for shared events between these two zip codes, as we expect anomalies to be shared across zip codes as well, especially if they originate at the middle-mile or beyond.

Table 2 shows the distribution of probe counts by ISP. The devices are connected to seven distinct ISPs, but our sample is dominated by Comcast and AT&T, which are the two most prominent ISPs in the area. The remaining ISPs are either smaller or provide specialized services, such as campus

networks or fixed wireless access. In Section 4.3, we analyze the impact of devices being in the same ISP on shared network anomalies.

Figure 1a shows the distribution of the number of latency measurements per probe. We use the last-mile latency measurement counts to construct this plot, and expect similar numbers for other targets as well. We see that more than 78% of devices have more than 10,000 measurements, with a median of 22,454 measurements per probe, per target. Additionally, we notice that two devices contain only 171 and 108 measurements, respectively, and do not register any anomalies within less than one day of deployment. We exclude these devices from our analysis, making our final dataset size 97 devices. We are unable to precisely determine the reason for the low measurement counts due to the lack of metadata, but we speculate that these devices were deployed for a short duration. Overall, the remaining devices offer a rich set of measurements, all of which may not perfectly align in time, but are still useful for our analysis.

Next, we assess typical speeds for each probe based on NDT speed test measurements, which we conduct at an hourly frequency. Figure 1b shows the distribution of median download speed across devices. We observe a step-like distribution, with each step likely signifying distinct speed tiers, similar to distinct clusters observed in crowdsourced datasets in prior work [73]. About 7% of devices have a median download speed of less than 50 Mbps, over 43% above 500 Mbps and nearly 15% above 900 Mbps, suggesting a good mix of devices with different speed tiers. The aggregate median download speed across all devices is about 347 Mbps, which is close to the typical speed observed by the FCC MBA program [27] around the same time period as our deployment [26]. Overall, we observe a range of speeds across our devices, suggesting that our analysis is not biased towards any particular speed tier.

We also look at the minimum last-mile latency distribution for the devices. We use the first public hop outside the home network as the hop for measuring last mile latency. Our last mile test involves sending traceroute probes to a fixed target, followed by grabbing the first public hop from the output. Then, we measure the round-trip time to this hop using ICMP echo requests. Figure 1c displays the distribution of minimum last mile latency across devices. We observe that a majority of devices (> 80%) have a last mile latency of less than 10 ms, which is indicative of fiber and cable connections. Nearly 10% of devices have a last mile latency of 10-20 ms, which is due to the presence of DSL connections in the sample; over 15 devices showed download speeds below 50 Mbps. Overall, our dataset contains a diverse set of devices with a mix of access technologies that we typically observe in residential networks.

While our sample is not representative of the entire U.S. population, it does provide a diverse set of devices connected to different ISPs, speed tiers and access technology types in an urban area. This diversity allows us to capture a broad range of real-world network conditions and performance behaviors, making our dataset suitable for analysis of shared network anomalies.

4 Spatio-temporally Correlated Latency Anomalies

In this section, we present our methodology for detecting shared latency anomalies in time and space. We begin by describing our approach to detect latency anomalies using a modified version of the Jitterbug congestion inference framework [15]. We then analyze the temporal overlap of shared events of elevated latency between probe pairs, and characterize the relationship between the similarity in amplitude and the temporal overlap of these events. Finally, we assess the impact of shared events of elevated latency.

4.1 Anomaly Detection

We make a number of design changes to the original Jitterbug methodology for our use case. First, the authors of the original work report that the BCP algorithm takes a longer time to process large

datasets (60-90 seconds for 15 days of data). Since our dataset contains a total of 14.6 million latency measurements from over 4 months of deployment, we extend the Jitterbug pipeline to support a more lightweight yet effective algorithm, called Pruned Exact Linear Time (PELT) [45]. PELT is based on a linear time dynamic programming algorithm that detects multiple change-points in a time-series by minimizing a cost function. We use PELT with an L2 cost function for our detections. The algorithm uses a penalty parameter to control the sensitivity of the detections. A higher penalty leads to the detection of fewer change-points. Since our objective is to detect overlaps, we set this parameter to 0.001 ms^2 to detect maximum, albeit insignificant, changes in latency. This results in a highly fragmented set of change points.

The change points obtained through PELT are then filtered and merged to identify significant latency jumps. So the second change we make is to modify the heuristics and filters used to detect jumps in latency. While continuing with Jitterbug's original threshold of 0.5 ms to infer a mean shift, we notice that the original approach infers both these cases as jumps: (1) a legitimate jump in latency from baseline, and (2) recovery to baseline after a "dip" in latency. We take a number of steps to avoid this behavior. First, we call a given segment a dip if its mean is lower than the mode of the entire signal (our measure of baseline). Second, we track the index of both the last detected dip and the last detected jump. Finally, we mark a segment as a jump only if: (1) its mean is 0.5 ms higher than the previous segment, (2) its mean is higher than the baseline, and (3) the last segment was not a dip.

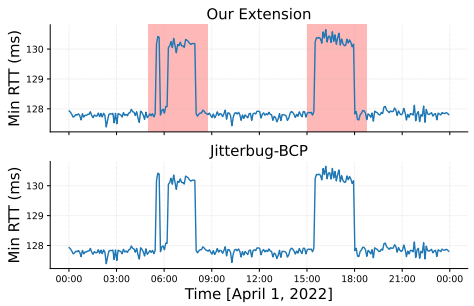


Fig. 2. Visual comparison between our extension and Jitterbug-BCP.

The third and final change we make is to modify the memory rule Jitterbug uses to avoid labelling the offset of a jump, i.e., the post-jump period of time when latency is on its way back to the baseline, as a dip. The original rule states that a change-point C_2 is also labelled as a jump if the previous change-point C_1 is a jump and the mean latency of C_2 is at least as large as that of C_1. Our modification is to check whether the absolute difference in maximum latency of adjacent segments is within 1.5 times the standard deviation of the entire signal, and label the second segment as a jump if the first segment is also a jump. Figure 2 shows a visual example for a probe's latency to Stockholm after applying these changes. We observe that our heuristics allow for precise detection while the change point boundaries go undetected using existing Jitterbug heuristics.

Our extensions to Jitterbug and the resulting pipeline are summarized in Figure 3. As in the original implementation, we first convert the raw RTT time series to a min-RTT series, with the min calculated in 15-minute bins. Then, we look for changes in the ISP of the probe based on the WHOIS lookups of the IP address used for each measurement. This is done to ensure that we are only looking at latency changes that are likely to be caused by a network event and not by a change enforced by the user. We find that four users switched their ISP without notification during our deployment, and we take measures to re-label and segment their measurements accordingly. Next, we sample the min-RTT series using sliding time windows with length 48 hours. We move the window by 24 hours at a time to ensure that the detected jumps are not located at the edges of the window. Using larger windows to detect long-term jumps is a ripe avenue for future work. In steps 4 and 5, we apply our modified Jitterbug methodology to detect anomalies in latency. In the final step, we compare these jumps across devices to compute overlaps in time. We deem two anomalies $E_1(s_1, e_1)$ and $E_2(s_2, e_2)$ to be overlapping if $s_1 < e_2$ and $s_2 < e_1$, where s_i and e_i are the start and end times of the anomalies, respectively. These overlaps are calculated uniquely at the level of individual device pairs and the latency destination.

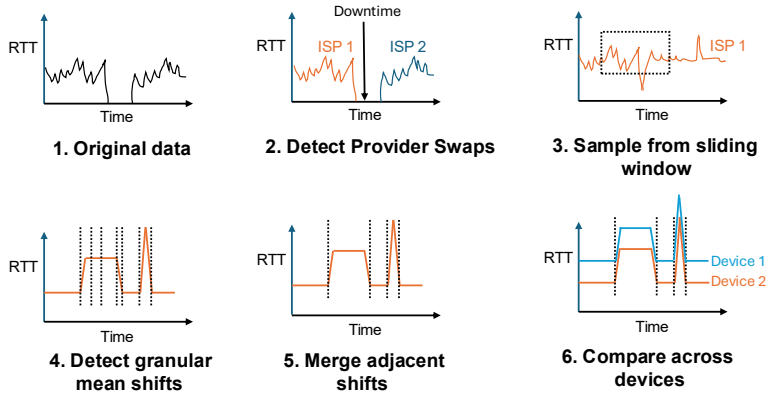


Fig. 3. An overview of our anomaly detection methodology. We detect mean shifts with sensitive parameters for every provider change in the dataset. After merging adjacent shifts, we analyze the co-occurrence of detected anomalies across devices located in the same geography.

4.2 Empirical Comparison with Jitterbug

To validate the effectiveness of our modifications to Jitterbug, we empirically compare the results of our modified approach with Jitterbug's original implementation. In the absence of ground truth data, we evaluate our approaches based on a synthetic dataset. Our synthetic dataset generation involves injecting controlled anomalies into RTT samples drawn from distributions that mimic our original dataset. Finally, we run both the original and modified detectors on the synthetic dataset and compare their outputs in terms of precision, recall and F1-score. Next, we describe the details of our evaluation and the results of our comparison.

For each probe-destination pair in our original dataset, we consider the use of IQR to identify the baseline range of RTT values. Specifically, we identify the baseline range of RTT values as $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$, where $Q1$ and $Q3$ are the first and third quartiles of the RTT values, respectively, and $IQR = Q3 - Q1$. In the next step, we generate synthetic RTT samples by drawing from a normal distribution with mean and standard deviation equal to the mean and standard deviation of the original RTT values that fall within the baseline range. For injecting anomalies, we consider sampling from a 2-state Markov chain to determine the state of the system (normal or anomalous) in a given time window. An alternative approach is to sample the state from a Bernoulli distribution. This approach assumes that anomalies occur independently across time windows, which may not be realistic, given significant temporal correlations in network data [48, 74]. The transition probabilities for the Markov chain are given by: $p_{\text{stay}} = 1 - \frac{1}{L}$, $p_{\text{enter}} = \frac{\rho}{L(1-\rho)}$, where L is the average duration of an anomaly in terms of the number of time windows, p_{stay} is the probability of staying in the anomalous state, and p_{enter} is the probability of entering the anomalous state. We derive these probabilities in Appendix B.

We set L to 20, which is equivalent to a duration of 5 hours, and we vary ρ from 0.05 to 0.5 to evaluate the performance of the detectors under different anomaly rates. Further, to control for the amplitude of the anomalies, we use a signal-to-noise ratio (SNR) parameter SNR_{min}, defined as the ratio between the standard deviation of the baseline RTT values and the mean shift in RTT values during an anomaly. We use an SNR_{min} of 1, which corresponds to a noise level of one standard deviation, enabling us to produce more challenging synthetic data for comparison. We analyze the impact of this parameter in Appendix C. Using these settings and after assigning the labels (0 for normal, 1 for anomalous) using 10 random seeds, we generate a synthetic dataset of 1000 time steps for all probe-destination pairs in our original dataset.

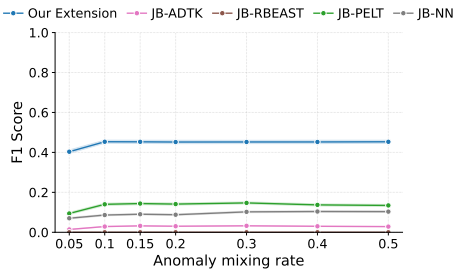


Fig. 4. F1-score comparison between methods.

At the time of writing, Jitterbug's implementation supports the following algorithms: (1) Bayesian Change Point detection (BCP), (2) PELT, (3) Rbeast [94], (4) rule-based methods implemented in Anomaly Detection Toolkit (ADTK) [5], and (5) Neural network-based detection. We are unable to run our complete synthetic dataset through BCP due to its long processing time. We run the remaining four algorithms on our synthetic dataset and compare their outputs with our modified approach. Figure 4 shows the F1-score of each method as a function of the anomaly mixing rate ρ. We observe that our modified approach outperforms the other methods across all anomaly rates. The average F1-score of our approach across all ρ values is 0.45, compared to 0.13 for the best performing Jitterbug baseline (PELT). Given that our data is class imbalanced (anomalies are rare), an F1 score of 0.45 is a significant improvement over the baselines. We also find that the performance of the Rbeast baseline is particularly poor, with an average F1-score of zero across all ρ values. Rbeast is designed to detect strong structural breaks in time-series data, which does not align well with our SNR_{min} setting of 1.

4.3 Analysis of Shared Anomalies

In this section, we characterize the overlap of shared events of elevated latency between probe pairs for a given destination across time and space. To quantify the extent to which anomalies are shared over time, we compute the intersection over union (IoU) based on their durations. For anomalies $E_1(s_1, e_1)$ and $E_2(s_2, e_2)$, the IoU is defined as $IoU(E_1, E_2) = \frac{max(0, min(e_1, e_2) - max(s_1, s_2))}{max(e_1, e_2) - min(s_1, s_2)}$. Here s_i and e_i are the start and end times of E_i, respectively. The IoU value ranges from 0 to 1, where 0 indicates no overlap and 1 indicates complete overlap. We compute the IoU for each pair of anomalies observed toward the same destination by a pair of devices.

We begin by analyzing the distribution of IoU of shared events of elevated latency. Figure 5 shows a CDF of the IoU values observed over a dataset of over 1.5 million paired events of elevated latency toward M-Lab servers. We observe that over 23% of overlapping events show an IoU of 0.8 or higher, and over 14% of overlapping events exhibit an IoU of 0.99 or higher. To validate that this behavior is not an artifact of the stochastic nature of network data or imperfect change-point detection, we conduct a randomization test. Specifically, we randomly shuffle the timestamps of the shared events within a day of their occurrence individually for each probe while preserving their duration and amplitude. We then recompute the IoU for the shuffled events for 1000 repetitions of the randomization to obtain a null distribution over the IoU values. In the null distribution, we observe only 1.458% ± 0.009% of overlapping events to show an IoU of 0.8 or higher. This indicates that the observed IoU values are significantly higher than what would be expected by chance, confirming that a significant number of events of elevated latency are correlated in time.

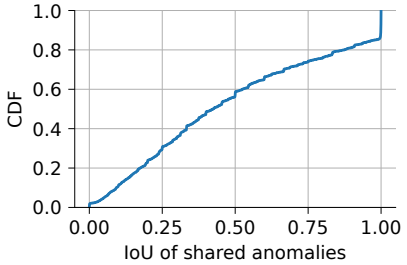
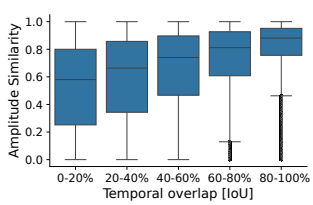
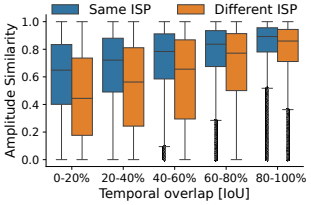


Fig. 5. Distribution of the IoU of shared events of elevated latency. Over 14% of overlapping events exhibit an IoU of 0.99 or higher.

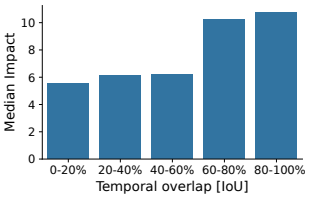
Next, we examine the relationship between the similarity in amplitude and the temporal overlap for shared latency events. We compute the amplitude as the difference between the maximum latency



(a) Overall amplitude similarity relationship with IoU.



(b) Amplitude similarity versus IoU stratified by shared or non-shared ISP.



(c) Median impact (ms-hours) versus IoU.

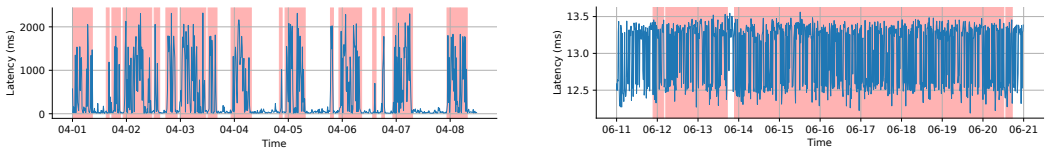
Fig. 6. Events with similar amplitudes often exhibit greater overlap in duration, indicating they likely reflect the same underlying phenomenon. This relationship is more pronounced when the events occur within the same ISP. Moreover, events with higher temporal overlap (IoU) tend to be the ones with greater impact.

during an anomaly and the baseline latency. Under the assumption that latency anomalies are rare events, we use the mode of the minimum latency (binned in 15-minute intervals) as the baseline latency. Duration, on the other hand, is defined as the difference in hours between the start and end times of the anomaly. For a pair of shared events of elevated latency, we define *amplitude similarity* as the ratio between the minimum and maximum amplitude of the shared events.

Figure 6a shows the overall amplitude similarity relationship with IoU. We observe that the similarity in amplitude increases with the IoU of the shared anomalies, with a median amplitude similarity of 0.88 for the 80-100% IoU range. This indicates that events with a high temporal overlap are also more likely to be similar in terms of their amplitude. These findings suggest that temporal alignment can be a strong indicator of underlying shared causes across measurement points, and that amplitude similarity can be used to attributing events to a common root cause. We also stratify the events by whether they are shared by the same ISP or not. Figure 6b shows the amplitude similarity versus IoU for events shared by the same ISP and those that are not. We observe that the correlation between amplitude similarity and IoU is slightly stronger for events shared by the same ISP, with a median amplitude similarity of 0.89 for the 80-100% IoU range, compared to 0.85 between ISPs. This indicates that events shared by the same ISP are slightly more likely to be similar in terms of their impact. The marginal difference in the correlations might be attributed to the off-net nature of M-Lab servers, where performance is more influenced by the path to the server than the last-mile ISP. We also note that the overall relationship between amplitude similarity and the IoU is not perfectly linear (Spearman's correlation coefficient = 0.37) due to the presence of outliers in the data. Since our change-point detection approach is applied individually to each probe, it is possible to get different amplitude values and imperfect IoUs for the same event across devices due to noise in the data.

We next assess the IoU ranges that are most likely to be associated with a high impact. We define the *impact* of events of elevated latency (in units of ms-hours) as the product of their amplitude and duration. We compute the median impact for each 20% IoU bin. Figure 6c shows the median impact (ms-hours) of shared events of elevated latency as a function of the 20% IoU bin they belong to. We observe that the median impact of shared events increases with the IoU, with the 60-100% IoU ranges showing a median impact of nearly 10 ms-hours. We note that a 10 ms-hour impact corresponds to the 72-percentile of the overall impact distribution, indicating that events showing a high mutual overlap tend to be more impactful than a typical event of elevated latency.

It is worth noting that outliers in both amplitude similarity and impact values can potentially affect our analysis. Some reasons for such scenarios could be merging of multiple tiny jumps in latency into a single large anomaly, bufferbloat, or due to a malfunctioning probe/home router. An example of two such cases is shown in Figure 7. In Figure 7a, we observe a large amplitude for the anomaly,



(a) A likely case of bufferbloat; our approach detects a large amplitude for the anomaly.

(b) A case with high variation in baseline latency; our approach leads to overestimation of duration.

Fig. 7. Examples of outliers in impact values. Red shaded regions denote detected anomalies.

which is likely due to bufferbloat or a malfunctioning probe. We also observe that this probe shows an atypical distribution of overall IoU values, with only 5% of the shared events showing an IoU of 0.8 or higher. In Figure 7b, we observe a case with oscillatory behavior in baseline latency between the 12.5-13.5 ms range. Since our methodology involves marking any jump with a higher than 0.5 ms mean shift as an anomaly, we observe many small jumps in latency being merged into a single large anomaly during the de-duplication process. This results in a large impact value of 168.1 ms-hours for the anomaly between June 13 and June 21. This particular anomaly shows a maximum IoU score of only 0.52 across all its shared events because of its large duration, suggesting that most of the overlaps associated with this probe are spurious. In Section 5.2, we discuss how our greedy algorithm for probe selection can be made robust to such outliers.

5 Data Reduction through Probe Selection

In this section, we propose an algorithm to select a subset of probes that maximizes the coverage of high-impact latency anomalies across the dataset. Our ultimate goal is to reduce measurement cost and redundancy while ensuring that we cover the most impactful network events. We first formulate the problem as a maximum weighted set coverage problem, which is known to be NP-hard. We then present a greedy approximation algorithm with provable guarantees, followed by an empirical evaluation on our dataset.

5.1 Problem Formulation

Let $\mathcal{P}$ denote the set of all probes deployed, and let $\mathcal{E}$ be the set of high-impact latency anomalies observed in the dataset. Each anomaly $e_k \in \mathcal{E}$ is associated with an impact score I_k, defined as the product of its amplitude and duration. Let $\mathcal{E}(p_i) \subseteq \mathcal{E}$ denote the set of anomalies observed by probe $p_i \in \mathcal{P}$. The goal is to find the smallest subset of probes $\mathcal{P}' \subseteq \mathcal{P}$ such that the total impact of the anomalies they collectively observe satisfies $\sum_{e_k \in \bigcup_{p_i \in \mathcal{P}'} \mathcal{E}(p_i)} I_k \geq c \cdot \sum_{e_k \in \mathcal{E}} I_k$. Here, $c \in (0, 1]$ is a user-defined coverage threshold (e.g., $c = 0.65$ for 65% impact coverage) to control the trade-off between minimizing the number of probes and maximizing coverage of impactful events. Since the absolute impact values I_k are subject to noise from change-point detection, it is likely that the impact of selected probes may exceed the coverage threshold with only a few selected probes. We verify this using pilot experiments, where using the absolute impact values led to overestimation of coverage using a limited number of probes. To minimize this effect, we replace I_k with its log-transform, $\log(1 + I_k)$ in the above inequality. This transformation shrinks the original range of impact values, making probe selection less sensitive to noise from change point detection while preserving the relative ordering of anomalies. We call this value *log-impact*.

5.2 Algorithm & Implementation

Before designing an algorithm to select the minimum number of probes, it is critical to assign a unique identifier to each anomaly e_k in the dataset. This helps us determine the set of probes that observe an anomaly due to the same network perturbation. Guided by previous discussions, we assign the same

unique identifier to a pair of anomalies if they share the same destination and overlap significantly in time (IoU $\geq \delta_{IoU}$). It is important to note that the choice of δ_{IoU} can significantly affect the total number of unique identifiers assigned to the anomalies. More relaxed values lead to more unique identifiers, which ultimately increases the total number of probes needed to cover the same set of anomalies. Further, the choice of change-point detection approach can also affect the number of unique identifiers assigned to the anomalies as there could be imperfect durational overlaps between devices. In Section 5.3, we experiment with different values of δ_{IoU} and the coverage fraction c, to find the best trade-off between minimizing the number of probes and maximizing coverage of impactful events. We originally considered using amplitude similarity (δ_{sim}) as an additional parameter for assigning unique identifiers, but chose to exclude it for two reasons: (1) our analysis visualized in Figure 6b suggests that barring few outliers, temporal overlap correlates with amplitude similarity and (2) relying on a single parameter reduces complexity and can provide better explainability in auditing our algorithm. Overall, we assign 53,110 unique identifiers to the anomalies, which are observed by 97 probes across all targets. Table 3 shows the breakdown of these anomalies by amplitude and duration bins.

	Duration (hours)				
Amplitude (ms)	<2	2–4	4–12	12–24	>24
0–10	18,516	15,129	10,960	2,428	1,041
10–25	499	440	438	125	128
25–50	226	205	198	41	13
50–100	190	132	92	22	8
>100	833	691	621	127	7

Table 3. No. of anomalies by amplitude and duration bins after assigning unique IDs ($\delta_{IoU} = 0.9$).

Our algorithm iteratively selects the probe that maximizes the marginal gain in coverage until the desired coverage threshold is met. Algorithm 1 outlines this approach. The algorithm maintains a set of selected probes $\mathcal{P}'$ and a set of covered anomalies $\mathcal{S}$. In each iteration, it selects the probe p^* that maximizes the sum of log-impact scores of the anomalies that are not already covered by the selected probes. The algorithm continues until the cumulative log-impact C of the covered anomalies reaches a fraction c of the total log-impact T.

The overall time complexity of this algorithm is $O(n^2 \cdot m)$, where n is the number of probes and m is the number of unique anomalies. We believe this is acceptable for most practical applications. This problem formulation and algorithm can be proven to achieve a guaranteed approximation of the optimal solution for a given probe budget. We refer an interested reader to relevant literature for more details on the exact approximation ratio and proof [65].

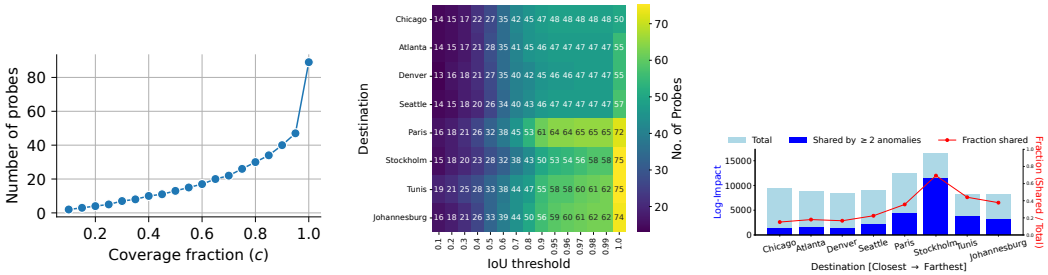
Algorithm 1 Greedy Probe Selection

$\mathcal{P}' \leftarrow \emptyset, \mathcal{S} \leftarrow \emptyset, C \leftarrow 0$
$T \leftarrow \sum_{e_k \in \mathcal{E}} \log(1 + I_k)$
while $C < c \cdot T$ **do**
  $p^* \leftarrow \arg\max_{p_i \in \mathcal{P} \setminus \mathcal{P}'} \sum_{e_k \in \mathcal{E}(p_i) \setminus \mathcal{S}} \log(1 + I_k)$
  $\mathcal{P}' \leftarrow \mathcal{P}' \cup \{p^*\}$
  $\mathcal{S} \leftarrow \mathcal{S} \cup \mathcal{E}(p^*)$
  $C \leftarrow \sum_{e_k \in \mathcal{S}} \log(1 + I_k)$
end while
return $\mathcal{P}'$

5.3 Empirical Evaluation

We now empirically evaluate the performance of our probe selection algorithm on latency measurements from the dataset described in Section 3.1. First, we evaluate the relationship between choices of the coverage fraction c, and the total number of probes selected with fixed values of δ_{IoU}. We then evaluate the effect of varying the threshold, δ_{IoU} on the number of probes selected. Finally, we compare our algorithm with baseline approaches, such as uniform random sampling and a naïve baseline that selects probes based on the descending order of their impact scores.

To evaluate the trade-off between coverage fraction and the number of selected probes, we fix $\delta_{IoU} = 0.9$ and vary the coverage fraction c from 0.1 to 1.0 in increments of 0.05. The choice of δ_{IoU} is motivated by our results from Figure 6c, which shows that a significant fraction of high impact anomalies have an IoU ≥ 0.6. We pick $\delta_{IoU} = 0.9$ as it ensures that there is a tight overlap between anomalies detected by different probes. Figure 8a shows the number of probes selected as a function of



(a) Number of probes vs. coverage fraction for an M-Lab server in Chicago.

(b) A heatmap of number of selected probes (out of 97) vs δ_{IoU} across destinations.

(c) Observed vs total log impact ($\delta_{IoU} = 0.9$).

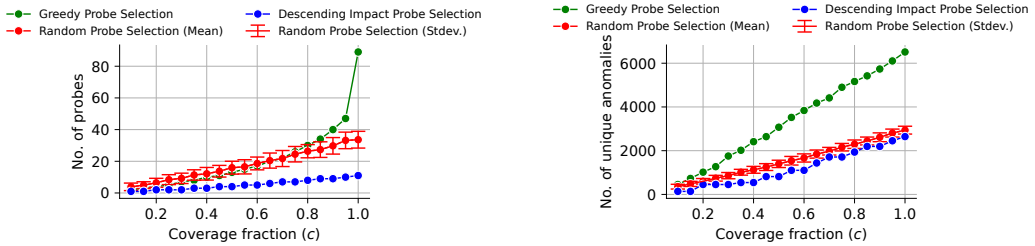
Fig. 8. We require a greater number of probes to achieve 95% impact coverage with higher values of δ_{IoU}. The fraction of shared log-impact increases with distance until Stockholm, but drops for farther targets.

the coverage fraction. As expected, the number of selected probes increases with increasing coverage fraction. Out of a total of 97 probes used in the deployment, the number of selected probes is 1 when $c = 0.1$, 47 when $c = 0.95$, and sharply increases to 89 when $c = 1.0$. Only 89 probes are selected out of 97 when $c = 1.0$ because the algorithm is able to cover nearly all the impact of latency anomalies, suggesting that the remaining probes do not contribute any new anomalies based on the threshold used. The number of probes selected increases slowly until $c = 0.95$, and provides marginal gains in coverage after this point, suggesting that the algorithm is able to cover a significant portion of the total impact with less than half the probes. We observe a similar trend for other latency destinations, with greater number of probes needed for destinations farther away from the probe deployment location to achieve 95% impact coverage. This is likely because of longer paths taken by packets to reach these destinations, leading to more diverse network perturbations being observed.

To evaluate this effect, we fix the coverage fraction to 0.95 and vary the anomaly temporal overlap threshold δ_{IoU} from 0.1 to 1.0 in increments of 0.1. For each value, we run Algorithm 1 and record the number of probes required to meet the coverage criterion. Figure 8b shows the resulting probe counts across multiple targets. As δ_{IoU} increases, the number of selected probes grows, reaching a maximum of 75 at $\delta_{IoU} = 1.0$, as stricter overlap reduces anomaly grouping and redundancy. Conversely, relaxing the threshold to 0.1 reduces the probe count to 13 by increasing overlap among similar events. Values between 0.90 and 0.99 provide low variability in probe counts across destinations, with a dramatic increase in probe counts at $\delta_{IoU} = 1.0$ due to the lack of any grouping. We recommend a value of $\delta_{IoU} = 0.9$ for this dataset, as it allows some flexibility in change point detection while ensuring that only highly similar events are grouped together.

We also observe significant destination-specific variation. This variation is driven by the fraction of log-impact shared across probes and by the effect of distance on anomaly detectability. Figure 8c shows that shared log-impact increases with distance and peaks at Stockholm, likely due to common upstream infrastructure. For more distant targets (e.g., Tunis and Johannesburg), total detected impact decreases due to fewer detectable anomalies, as larger baseline RTTs and path diversity make small latency increases harder to detect, which leads us to select more probes for comparable coverage.

We compare our probe selection algorithm against two baselines: (1) uniform random sampling until the coverage threshold is met, and (2) selecting probes in descending order of log-impact without accounting for overlap. We evaluate performance using the number of probes required to achieve the target coverage and by the number of unique anomalies observed. We repeat random sampling 100 times and average the results. We evaluate all methods using a Chicago-based M-Lab server as the destination and set $\delta_{IoU} = 0.9$. Figure 9 summarizes the results, with error bars indicating



(a) Probe counts across methods. (b) Number of unique anomalies across methods.

Fig. 9. A comparison of our approach with two baseline approaches. Our approach covers significantly higher number of unique anomalies compared to the baselines, while selecting a comparable number of probes to random sampling.

standard deviation for random sampling. Our approach selects a similar number of probes as random sampling and more probes than the descending log-impact baseline to achieve comparable coverage (Figure 9a), but consistently captures significantly more unique anomalies across nearly all coverage levels (Figure 9b). This indicates that our method better prioritizes diversity in high-impact anomalies, whereas the baselines tend to select redundant probes. This trend persists for geographically farther targets.

6 Predictive Evaluation

Our approach thus far has focused on a greedy set cover based post-hoc optimization of probe selection to reduce redundancy in detected anomalies. We started with a non-random deployment of probes in a single urban area and showed that we can reduce the number of probes needed to capture a high fraction of the total anomaly impact. This finding can readily allow for dimensionality reduction of datasets collected from existing probe deployments that collect end-to-end delay measurements (e.g., FCC MBA [27], RIPE Atlas [80], and M-Lab [61]). However, a more interesting question is to ask: *how well does the reduced set of probes predict future anomalies?* In this section, we perform a predictive evaluation of our approach to answer this question. We also consider analyzing the geographic distribution of the reduced probe set in Appendix D.

To assess whether our approach selects probes that continue to cover a large fraction of anomalies over time, we perform a predictive evaluation using a time-based split of the dataset. Specifically, we use the first T days to select probes via our greedy heuristic, and evaluate the fraction of anomalies covered by these probes in the remaining days. We vary T from 7 to 56 days in increments of 7 days. For each T, probe selection is re-optimized using only data from the first T days. We conduct this evaluation for multiple targets within the largest ISP in our dataset (Comcast). To account for probe churn, we restrict our focus to probes active in both the training and evaluation periods, resulting in 29 eligible probes. For each T, we apply our greedy heuristic to this set and report anomaly coverage in the subsequent period.

Table 4 shows the recall scores, i.e., fraction of anomalies covered in the future period by probes selected using data from the selection window for different values of T. We exclude precision because it does not provide useful information since there is no notion of false positives in this setting. A similar argument applies for accuracy or F-1 score as they also rely on false positives. We observe that the recall scores follow a similar trend as the fraction of shared log-impact in Figure 8c. For example, the recall scores peak at Stockholm, and finally drop for the farthest targets (Tunis and Johannesburg). This suggests that the fraction of shared log-impact between probes and targets is a good predictor of how well our approach can select probes that cover future anomalies. In other words, the ability of a small-scale probe deployment to generalize to future anomalies depends on

Selection Window (# days)	M-Lab Server Location (Increasing order of distance →)															
	CHI		ATL		DEN		SEA		PAR		STO		TUN		JOH	
	R	C	R	C	R	C	R	C	R	C	R	C	R	C	R	C
7	**0.72**	17	**0.80**	15	**0.71**	16	**0.87**	14	**0.91**	19	**0.94**	9	**0.93**	12	**0.92**	17
14	**0.74**	19	**0.82**	15	**0.72**	16	**0.86**	15	**0.89**	19	**0.96**	13	**0.94**	14	**0.92**	17
21	**0.74**	19	**0.82**	16	**0.73**	17	**0.88**	16	**0.87**	19	**0.97**	14	**0.95**	15	**0.92**	18
28	**0.73**	19	**0.75**	17	**0.71**	17	**0.85**	17	**0.85**	19	**0.96**	15	**0.94**	16	**0.93**	19
35	**0.71**	19	**0.74**	17	**0.69**	17	**0.81**	17	**0.84**	19	**0.96**	16	**0.94**	16	**0.93**	19
42	**0.70**	19	**0.74**	17	**0.68**	17	**0.76**	17	**0.83**	20	**0.97**	17	**0.94**	16	**0.94**	19
49	**0.70**	19	**0.75**	17	**0.67**	17	**0.75**	16	**0.80**	20	**0.96**	17	**0.92**	16	**0.94**	19
56	**0.70**	19	**0.76**	17	**0.67**	17	**0.75**	16	**0.76**	19	**0.96**	17	**0.93**	16	**0.95**	20

Table 4. Prediction performance across different targets as a function of selection window length for Comcast probes. Target locations are abbreviated by their first three letters. For each location, values in the **R** column denote recall while values in the **C** column denote probe counts. We observe invariance in recall scores across different selection window lengths. Additionally, for a given selection window, the recall scores follow a similar trend as the fraction of shared log-impact in Figure 8c.

the location of the target being measured, with targets that are neither too close nor too far away providing the best generalization performance.

Table 4 also tells us that the length of the selection window T does not have a significant impact on the recall scores. This suggests that even a small amount of historical data can be sufficient to select probes that continue to provide a steady level of coverage of anomalies in the future. This is a promising finding, potentially enabling operators, researchers, or regulators who start with a small set of probes to experiment with multiple deployment strategies, collect data for a short period of time, and then use our approach to select a smaller subset of probes for long-term deployment. An alternative approach could also be to start with a few probes and incrementally add more to the deployment until a desired recall score is achieved. We leave the exploration of such approaches to future work.

7 Discussion, Limitations & Future Work

In this work, we apply change-point detection to longitudinal idle latency measurements and propose a sampling algorithm that selects a representative probe set while minimizing redundant detections. While our results provide a useful starting point for leveraging shared anomalies for probe selection, we discuss implications, limitations, and future directions of our approach below.

Our approach provides post-hoc analysis of shared latency anomalies in a given deployment of probes in a city. Existing deployments of longitudinal measurement platforms such as RIPE Atlas [80] and FCC's Measuring Broadband America (MBA) [27] can readily benefit from our approach by using it to select probes for future measurements. We also urge stakeholders, especially measurement application developers (e.g., Ookla [66] and M-Lab [61]), to consider reporting amplitudes and durations of idle latency anomalies, whenever they occur. As we have demonstrated through our approach, this would allow the discovery of geographies and timestamps that can benefit from additional data collection, as the current datasets only provide views of a cross-section of network performance at a given time from a limited number of devices [86].

While our approach does not have substantial resource overhead, improving its efficiency could make its adoption more appealing to measurement platforms. The primary bottleneck of our approach is computing anomalies and their overlaps, where given a deployment with n probes and d destinations, the runtime complexity is $O(n^2 \cdot d)$. To contextualize this, experiments run on our studied deployment (99 residential probes, eight destinations) typically had completion times of ~8.5 minutes (~0.64 seconds per probe/destination pair). In comparison, evaluating the Jitterbug-BCP approach on the

same deployment would take ~26.4 days (~50 minutes per probe/destination pair). One possible method towards further reducing the overhead of our approach is to employ hashing-based techniques (e.g., locality-sensitive hashing) for computing approximate, instead of explicit overlaps. Another could involve collecting fewer, more informative measurements that retain the ability to reconstruct or estimate metrics of interest (e.g., Nyquist-Shannon sampling). We plan on exploring the latter in future work.

While we believe that our methodology is generalizable to other geographies and platforms, we have not extensively verified it in other settings. This is important, as inaccurate change point detection and down-sampling can result in false-positives may misinform future decisions towards measurement, infrastructure deployment, and policymaking. Our analysis of the sampling methodology presented in this paper considers probe deployments measuring predominantly from two ISPs in a single city, over a four-month collection window. Indeed, the effectiveness of our sampling algorithm and anomaly detection pipeline depends on several contextual factors, including probe density, frequency of measurements, normal treatment of ICMP traffic, ISP share, and geographic diversity. We also do not provide comprehensive comparison of our strategy to existing ones, due to differences in deployment scenarios and prerequisite input data (e.g., Metis [4] uses data from RIPE Atlas, and recommends a minimum probe set size of 500 probes) that differ from our evaluated deployment scenario. Future work should thus explore the both utility of our sampling algorithm and comparison against existing strategies when spanning different probing methods (i.e., UDP/TCP-based) and regions, varying probe density, and different ISP-wise distribution of probes. Additionally, as latency anomalies can change overtime, it would also be valuable to understand if our approach is natively robust to these anomaly drifts, or if there is a need for periodic adaptation of the probe selection strategy.

8 Conclusion

In this work, we apply change point detection to longitudinal idle latency measurements to characterize network anomalies by their amplitude and duration, and propose a topology-free probe selection algorithm that reduces redundant detections.[2] In the absence of routing or path information, our approach exploits temporal overlap and similarity in anomaly magnitude across probes to identify a representative subset that preserves coverage of high-impact events. Our results show that shared anomalies provide a strong signal for distinguishing probes that contribute new information from those that observe largely redundant behavior.

Based on these findings, we recommend that measurement platforms explicitly detect, share and leverage information shared anomalies when selecting probes. Future work should validate these techniques across multiple deployments, improve anomaly detection precision, and develop sampling strategies that better account for additional operational constraints.

Acknowledgements

We thank our shepherd Philipp Richter and our anonymous reviewers for their feedback and suggestions. We are also grateful to members of the Internet Innovation Initiative at the University of Chicago for their efforts in developing, deploying and maintaining our measurement infrastructure. This work has been supported by a grant from the Agence Nationale de la Recherche (project no. ANR-24-CE25-1133 [GTTP]).

[2]The code and the dataset used in our paper is publicly available at https://github.com/noise-lab/probe-selection.

References

[1] IPinfo 2025. *IPinfo: IP Address API and Data.* IPinfo. https://ipinfo.io/

[2] Aditya Akella, Srinivasan Seshan, and Anees Shaikh. 2003. An empirical evaluation of wide-area internet bottlenecks. In *Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*. 101–114.

[3] Catalina Alvarez and Katerina Argyraki. 2023. Using Gaming Footage as a source of Internet latency information. In *Proceedings of the 2023 ACM on Internet Measurement Conference*. 606–626.

[4] Malte Appel, Emile Aben, and Romain Fontugne. 2022. Metis: Better Atlas Vantage Point Selection for Everyone.. In *TMA*.

[5] Arundo. 2020. ADTK: Anomaly Detection Toolkit. https://github.com/arundo/adtk.

[6] RIPE Atlas. 2025. DNSMON. https://dnsmon.ripe.net. Accessed: 2025-12-12.

[7] Vaibhav Bajpai, Steffie Jacob Eravuchira, and Jürgen Schönwä lder. 2015. Lessons learned from using the ripe atlas platform for measurement research. *ACM SIGCOMM Computer Communication Review* 45, 3 (2015), 35–42.

[8] Guillermo Baltra and John Heidemann. 2020. Improving Coverage of Internet Outage Detection in Sparse Blocks. In *Passive and Active Measurement*, Anna Sperotto, Alberto Dainotti, and Burkhard Stiller (Eds.). Vol. 12048. Springer International Publishing, 19–36. https://doi.org/10.1007/978-3-030-44081-7_2

[9] Guillermo Baltra, Tarang Saluja, Yuri Pradkin, and John Heidemann. 2023-10-26. *What Is The Internet? Partial Connectivity in the Internet Core.* https://doi.org/10.48550/arXiv.2107.11439 arXiv:2107.11439 [cs]

[10] Paul Barford, Azer Bestavros, John Byers, and Mark Crovella. 2001. On the marginal utility of network topology measurements. In *Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*. 5–17.

[11] Paul Barford, Jeffery Kline, David Plonka, and Amos Ron. 2002. A Signal Analysis of Network Traffic Anomalies. In *Proceedings of the 2nd ACM SIGCOMM Internet Measurement Workshop (IMW)*. 71–82.

[12] Francesco Bronzino, Nick Feamster, Shinan Liu, James Saxon, and Paul Schmitt. 2021. Mapping the digital divide: before, during, and after COVID-19. In *TPRC48: The 48th research conference on communication, information and internet policy.*

[13] Xue Cai and John Heidemann. 2010. Understanding Block-level Address Usage in the Visible Internet. (2010).

[14] Igor Canadi, Paul Barford, and Joel Sommers. 2012. Revisiting broadband performance. In *Proceedings of the 2012 Internet Measurement Conference*. 273–286.

[15] Esteban Carisimo, Ricky KP Mok, David D Clark, and Kimberly C Claffy. 2022. Jitterbug: A New Framework for Jitter-Based Congestion Inference. In *International Conference on Passive and Active Network Measurement*. Springer, 155–179.

[16] Josiah Chavula, Amreesh Phokeer, Agustin Formoso, and Nick Feamster. 2017. Insight into Africa's country-level latencies. In *2017 IEEE AFRICON*. IEEE, 938–944.

[17] Lechang Cheng, Norm C. Hutchinson, and Mabo R. Ito. 2008-03. RealNet: A Topology Generator Based on Real Internet Topology. In *22nd International Conference on Advanced Information Networking and Applications - Workshops (Aina Workshops 2008)*. 526–532. https://doi.org/10.1109/WAINA.2008.66

[18] Marshini Chetty, Srikanth Sundaresan, Sachit Muckaden, Nick Feamster, and Enrico Calandro. 2013. Measuring broadband performance in South Africa. In *Proceedings of the 4th Annual Symposium on Computing for Development*. 1–10.

[19] Cisco. 2020. Cisco Annual Internet Report (2018–2023) White Paper. (2020). Available at: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html.

[20] JURGEN CITO, Devan Gotowka, Philipp Leitner, Ryan Pelette, Dritan Suljoti, and Schahram Dustdar. 2015. Identifying web performance degradations through synthetic and real-user monitoring. *Journal of Web Engineering* (2015), 414–442.

[21] David D Clark and Sara Wedeman. 2022. Understanding the metrics of internet broadband access: How much is enough? *Available at SSRN 4178804* (2022).

[22] Xiaohong Deng, Yun Feng, Thanchanok Sutjarittham, Hassan Habibi Gharakheili, Blanca Gallego, and Vijay Sivaraman. 2021. Comparing Broadband ISP Performance using Big Data from M-Lab. *arXiv preprint arXiv:2101.09795* (2021).

[23] Shivani Deshpande, Marina Thottan, Tin Kam Ho, and Biplab Sikdar. 2009. An online mechanism for BGP instability detection and analysis. *IEEE transactions on Computers* 58, 11 (2009), 1470–1484.

[24] Amogh Dhamdhere, David D Clark, Alexander Gamero-Garrido, Matthew Luckie, Ricky KP Mok, Gautam Akiwate, Kabir Gogia, Vaibhav Bajpai, Alex C Snoeren, and Kc Claffy. 2018. Inferring persistent interdomain congestion. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*. 1–15.

[25] Mentari Djatmiko, Dominik Schatzmann, Xenofontas Dimitropoulos, Arik Friedman, and Roksana Boreli. 2013-11. Collaborative Network Outage Troubleshooting with Secure Multiparty Computation. 51, 11 (2013-11), 78–84. https://doi.org/10.1109/MCOM.2013.6658656

[26] Federal Communications Commission. 2023. *Measuring Fixed Broadband – Twelfth Report.* Technical Report. Federal Communications Commission, Washington, D.C. https://www.fcc.gov/reports-research/reports/measuring-broadband-america/measuring-fixed-broadband-twelfth-report Office of Engineering and Technology.

[27] Federal Communications Commission. 2025. Measuring Broadband America. https://www.fcc.gov/reports-research/reports/measuring-broadband-america. Accessed: 2025-05-11.

[28] Matt Fleming, Piotr Kolaczkowski, Ishita Kumar, Shaunak Das, Sean McCarthy, Pushkala Pattabhiraman, and Henrik Ingo. 2023. Hunter: Using Change Point Detection to Hunt for Performance Regressions. In *Proceedings of the 2023 ACM/SPEC International Conference on Performance Engineering*. 199–206.

[29] Roberto Gallardo and Brian Whitacre. 2024. An unexpected digital divide? A look at internet speeds and socioeconomic groups. *Telecommunications Policy* 48, 6 (2024), 102777.

[30] Jim Gettys and Kathleen Nichols. 2012. Bufferbloat: dark buffers in the internet. *Commun. ACM* 55, 1 (2012), 57–65.

[31] Andreas Guillot, Romain Fontugne, Philipp Winter, Pascal Merindol, Alistair King, Alberto Dainotti, and Cristel Pelsser. 2019-06. Chocolatine: Outage Detection for Internet Background Radiation. In *2019 Network Traffic Measurement and Analysis Conference (TMA)*. 1–8. https://doi.org/10.23919/TMA.2019.8784607

[32] John Heidemann, Yuri Pradkin, Ramesh Govindan, Christos Papadopoulos, Genevieve Bartlett, and Joseph Bannister. 2008-10-20. Census and Survey of the Visible Internet. In *Proceedings of the 8th ACM SIGCOMM Conference on Internet Measurement* (Vouliagmeni Greece). ACM, 169–182. https://doi.org/10.1145/1452520.1452542

[33] Thomas Holterbach, Emile Aben, Cristel Pelsser, Randy Bush, and Laurent Vanbever. 2017. Measurement vantage point selection using a similarity metric. In *Proceedings of the 2017 Applied Networking Research Workshop*. 1–3.

[34] Bingnan Hou, Changsheng Hou, Tongqing Zhou, Zhiping Cai, and Fang Liu. 2021. Detection and characterization of network anomalies in large-scale RTT time series. *IEEE Transactions on Network and Service Management* 18, 1 (2021), 793–806.

[35] Jiyao Hu, Zhenyu Zhou, and Xiaowei Yang. 2024. TelApart: Differentiating Network Faults from Customer-Premise Faults in Cable Broadband Networks. *arXiv preprint arXiv:2412.09740* (2024).

[36] Jiyao Hu, Zhenyu Zhou, Xiaowei Yang, Jacob Malone, and Jonathan W Williams. 2020. {CableMon}: Improving the reliability of cable broadband networks via proactive network maintenance. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*. 619–632.

[37] Zi Hu, John Heidemann, and Yuri Pradkin. 2012-11-14. Towards Geolocation of Millions of IP Addresses. In *Proceedings of the 2012 Internet Measurement Conference* (Boston Massachusetts USA). ACM, 123–130. https://doi.org/10.1145/2398776.2398790

[38] Hanyang Jiang, Henry Shaowu Yuchi, Elizabeth Belding, Ellen Zegura, and Yao Xie. 2023. Mobile Internet Quality Estimation using Self-Tuning Kernel Regression. *arXiv preprint arXiv:2311.05641* (2023).

[39] Daniel Jubas. 2021. Detecting BGP Interception Attacks using RTT Measurements.

[40] Ethan Katz-Bassett, Harsha V Madhyastha, John P John, Arvind Krishnamurthy, David Wetherall, and Thomas E Anderson. 2008. Studying Black Holes in the Internet with Hubble.. In *NSDI*, Vol. 8. 247–262.

[41] Ethan Katz-Bassett, Colin Scott, David R. Choffnes, Ítalo Cunha, Vytautas Valancius, Nick Feamster, Harsha V. Madhyastha, Thomas Anderson, and Arvind Krishnamurthy. 2012-09-24. LIFEGUARD: Practical Repair of Persistent Route Failures. 42, 4 (2012-09-24), 395–406. https://doi.org/10.1145/2377677.2377756

[42] Jerome L Kaufman. 2013. Chicago: segregation and the new urban poverty. In *Urban segregation and the welfare state*. Routledge, 45–63.

[43] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 137–146.

[44] Samir Khuller, Anna Moss, and Joseph (Seffi) Naor. 1999. The Budgeted Maximum Coverage Problem. *Inform. Process. Lett.* 70, 1 (1999), 39–45.

[45] Rebecca Killick, Paul Fearnhead, and Idris A Eckley. 2012. Optimal detection of changepoints with a linear computational cost. *J. Amer. Statist. Assoc.* 107, 500 (2012), 1590–1598.

[46] Andreas Krause, Ajit Singh, and Carlos Guestrin. 2008. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research* 9, 2 (2008).

[47] Hyeongseong Lee, Udit Paul, Arpit Gupta, Elizabeth Belding, and Mengyang Gu. 2023. Analyzing Disparity and Temporal Progression of Internet Quality through Crowdsourced Measurements with Bias-Correction. *arXiv preprint arXiv:2310.16136* (2023).

[48] Will E Leland, Walter Willinger, Murad S Taqqu, and Daniel V Wilson. 1995. On the self-similar nature of Ethernet traffic. *ACM SIGCOMM computer communication review* 25, 1 (1995), 202–213.

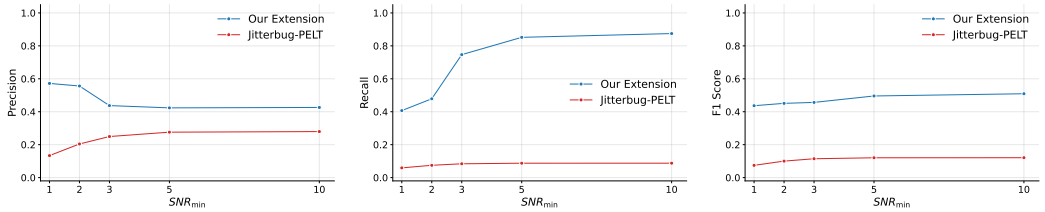
[49] Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. 2007. Cost-effective Outbreak Detection in Networks. In *Proc. 13th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*. 420–429.

[50] Hui Lin and Jeff Bilmes. 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*. 510–520.

[51] Simon Lindståhl, Alexandre Proutiere, and Andreas Johnsson. 2023. Change Point Detection with Adaptive Measurement Schedules for Network Performance Verification. 7, 3, Article 53 (Dec. 2023), 30 pages. https://doi.org/10.1145/3626784

[52] Matthew Luckie, Amogh Dhamdhere, David Clark, Bradley Huffaker, and KC Claffy. 2014. Challenges in inferring internet interdomain congestion. In *Proceedings of the 2014 Conference on Internet Measurement Conference*. 15–22.
[53] Kyle MacMillan, Tarun Mangla, James Saxon, and Nick Feamster. 2021. Measuring the performance and network utilization of popular video conferencing applications. In *Proceedings of the 21st ACM Internet Measurement Conference*. 229–244.
[54] Kyle MacMillan, Tarun Mangla, James Saxon, Nicole P Marwell, and Nick Feamster. 2023. A comparative analysis of ookla speedtest and measurement labs network diagnostic test (ndt7). *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 7, 1 (2023), 1–26.
[55] Haarika Manda, Varshika Srinivasavaradhan, Laasya Koduru, Kevin Zhang, Xuanhe Zhou, Udit Paul, Elizabeth Belding, Arpit Gupta, and Tejas N. Narechania. 2024. The Efficacy of the Connect America Fund in Addressing US Internet Access Inequities. In *Proceedings of the ACM SIGCOMM 2024 Conference (SIGCOMM '24)*. Association for Computing Machinery, New York, NY, USA, 484–505. https://doi.org/10.1145/3651890.3672266
[56] Tarun Mangla, Emir Halepovic, Mostafa Ammar, and Ellen Zegura. 2019. Using session modeling to estimate HTTP-based video QoE metrics from encrypted network traffic. *IEEE Transactions on Network and Service Management* 16, 3 (2019), 1086–1099.
[57] Tarun Mangla, Udit Paul, Arpit Gupta, Nicole P Marwell, and Nick Feamster. 2022. Internet inequity in Chicago: Adoption, affordability, and availability. *TPRC (August 5, 2022)* (2022).
[58] Jonatas Marques, Alexis Schrubbe, Nicole P. Marwell, and Nick Feamster. 2024. Are We Up to the Challenge? An analysis of the FCC Broadband Data Collection Fixed Internet Availability Challenges. arXiv:2404.04189 [cs.NI] https://arxiv.org/abs/2404.04189
[59] MaxMind. 2025. MaxMind GeoLite2 Geolocation Database. https://www.maxmind.com. Accessed: 2025-10-05.
[60] Conleth McCallan. 2023. A Brief History of Internet Speed. (2023). Available at: https://www.datanet.co.uk/brief-history-speed/.
[61] Measurement Lab (M-Lab). 2026. Measurement Lab (M-Lab). https://www.measurementlab.net. Accessed: 2025-05-11.
[62] Oliver Michel, Satadal Sengupta, Hyojoon Kim, Ravi Netravali, and Jennifer Rexford. 2022. Enabling passive measurement of zoom performance in production networks. In *Proceedings of the 22nd ACM internet measurement conference*. 244–260.
[63] Maxime Mouchet, Sandrine Vaton, and Thierry Chonavel. 2019. A flexible infinite HMM model for accurate characterization and segmentation of RTT timeseries. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 1055–1056.
[64] Syed Tauhidun Nabi, Zhuowei Wen, Brooke Ritter, and Shaddi Hasan. 2024. Red is Sus: Automated Identification of Low-Quality Service Availability Claims in the US National Broadband Map. In *Proceedings of the 2024 ACM on Internet Measurement Conference*. 2–18.
[65] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functions—I. *Mathematical programming* 14 (1978), 265–294.
[66] Ookla. 2025. Speedtest by Ookla. https://www.speedtest.net. Accessed: 2025-05-11.
[67] Ookla, Speedtest. 2025. Ookla's Open Data Initiative. https://www.ookla.com/ookla-for-good/open-data. https://www.ookla.com/ookla-for-good/open-data Accessed: 2025-12-12.
[68] Selim Ozcan, Ioana Livadariu, Georgios Smaragdakis, and Carsten Griwodz. 2023. Longitudinal Analysis of Inter-City Network Delays. In *2023 7th Network Traffic Measurement and Analysis Conference (TMA)*. IEEE, 1–9.
[69] Ramakrishna Padmanabhan, Aaron Schulman, Alberto Dainotti, Dave Levin, and Neil Spring. 2019. How to Find Correlated Internet Failures. In *Passive and Active Measurement*, David Choffnes and Marinho Barcellos (Eds.). Vol. 11419. Springer International Publishing, 210–227. https://doi.org/10.1007/978-3-030-15986-3_14
[70] Dimitrios Papakyriakou and Ioannis S Barbounakis. 2023. Benchmarking and review of raspberry pi (rpi) 2b vs rpi 3b vs rpi 3b+ vs rpi 4b (8gb). *International Journal of Computer Applications* 975, 3 (2023), 8887.
[71] Udit Paul, Vinothini Gunasekaran, Jiamo Liu, Tejas N Narechania, Arpit Gupta, and Elizabeth Belding. 2023. Decoding the divide: Analyzing disparities in broadband plans offered by major US ISPs. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 578–591.
[72] Udit Paul, Jiamo Liu, David Farias-Llerenas, Vivek Adarsh, Arpit Gupta, and Elizabeth Belding. 2022. Characterizing internet access and quality inequities in california m-lab measurements. In *Proceedings of the 5th ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies*. 257–265.
[73] Udit Paul, Jiamo Liu, Mengyang Gu, Arpit Gupta, and Elizabeth Belding. 2022. The importance of contextualization of crowdsourced active speed test measurements. In *Proceedings of the 22nd ACM Internet Measurement Conference*. 274–289.
[74] Vern Paxson and Sally Floyd. 2002. Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on networking* 3, 3 (2002), 226–244.
[75] Lin Quan, John Heidemann, and Yuri Pradkin. 2013. Trinocular: Understanding Internet Reliability Through Adaptive Probing. (2013).

[76] Lin Quan, John Heidemann, and Yuri Pradkin. 2014. When the Internet sleeps: Correlating diurnal networks with external factors. In *Proceedings of the 2014 Conference on Internet Measurement Conference*. 87–100.
[77] J Renita and N Edna Elizabeth. 2017. Network's server monitoring and analysis using Nagios. In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE, 1904–1909.
[78] Philipp Richter, Ramakrishna Padmanabhan, Neil Spring, Arthur Berger, and David Clark. 2018. Advancing the art of internet edge outage detection. In *Proceedings of the Internet Measurement Conference 2018*. 350–363.
[79] Philipp Richter, Georgios Smaragdakis, David Plonka, and Arthur Berger. 2016-11-14. Beyond Counting: New Perspectives on the Active IPv4 Address Space. In *Proceedings of the 2016 Internet Measurement Conference* (Santa Monica California USA). ACM, 135–149. https://doi.org/10.1145/2987443.2987473
[80] RIPE NCC. 2025. RIPE Atlas: A Global Internet Measurement Platform. https://atlas.ripe.net. Accessed: 2025-05-11.
[81] Swati Roy and Nick Feamster. 2013. Characterizing correlated latency anomalies in broadband access networks. In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*. 525–526.
[82] James Saxon and Dan A Black. 2022. What we can learn from selected, unmatched data: measuring Internet inequality in Chicago. *Computers, Environment and Urban Systems* 98 (2022), 101874.
[83] Ranya Sharma, Tarun Mangla, James Saxon, Marc Richardson, Nick Feamster, and Nicole P Marwell. 2022. Benchmarks or equity? A new approach to measuring internet performance. *A New Approach to Measuring Internet Performance (August 3, 2022)* (2022).
[84] Taveesh Sharma, Tarun Mangla, Arpit Gupta, Junchen Jiang, and Nick Feamster. 2023. Estimating webrtc video qoe metrics without using application headers. In *Proceedings of the 2023 ACM on Internet Measurement Conference*. 485–500.
[85] Taveesh Sharma, Jonatas Marques, Nick Feamster, and Nicole P Marwell. 2023. A First Look at the Spatial and Temporal Variability of Internet Performance Data in Hyperlocal Geographies. *Available at SSRN 4568668* (2023).
[86] Taveesh Sharma, Paul Schmitt, Francesco Bronzino, Nick Feamster, and Nicole P Marwell. 2024. Beyond Data Points: Regionalizing Crowdsourced Latency Measurements. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 8, 3 (2024), 1–24.
[87] Yuval Shavitt and Udi Weinsberg. 2009. Quantifying the importance of vantage points distribution in internet topology measurements. In *IEEE INFOCOM 2009*. IEEE, 792–800.
[88] Srikanth Sundaresan, Walter De Donato, Nick Feamster, Renata Teixeira, Sam Crawford, and Antonio Pescapè. 2011. Broadband internet performance: a view from the gateway. *ACM SIGCOMM computer communication review* 41, 4 (2011), 134–145.
[89] Srikanth Sundaresan, Walter De Donato, Nick Feamster, Renata Teixeira, Sam Crawford, and Antonio Pescapè. 2012. Measuring home broadband performance. *Commun. ACM* 55, 11 (2012), 100–109.
[90] Wayne A Taylor. 2000. Change-point analysis: a powerful new tool for detecting changes.
[91] Xiang Xuan and Kevin Murphy. 2007. Modeling changing dependency structure in multivariate time series. In *Proceedings of the 24th international conference on Machine learning*. 1055–1062.
[92] He Yan, Ricardo Oliveira, Kevin Burnett, Dave Matthews, Lixia Zhang, and Dan Massey. 2009. BGPmon: A real-time, scalable, extensible monitoring system. In *2009 Cybersecurity Applications & Technology Conference for Homeland Security*. IEEE, 212–223.
[93] Ming Zhang, Chi Zhang, Vivek Pai, Larry Peterson, and Randy Wang. 2004. PlanetSeer: Internet Path Failure Monitoring and Characterization in Wide-Area Services. (2004).
[94] Kaiguang Zhao, Michael A. Wulder, Tongxi Hu, Ryan Bright, Qiusheng Wu, Haiming Qin, Yang Li, Elizabeth Toman, Bani Mallick, Xuesong Zhang, et al. 2019. Detecting change-point, trend, and seasonality in satellite time series data to track abrupt changes and nonlinear dynamics: A Bayesian ensemble algorithm. *Remote Sensing of Environment* 232 (2019), 111181.



(a) Precision (b) Recall (c) F1-score

Fig. 10. Impact of the signal-to-noise parameter SNR_{min} on the precision, recall, and F1-score of detected anomalies against the best performing Jitterbug variant, JB-PELT.

A Ethics

Our deployment of Raspberry Pi probes in residential networks was approved by our Institutional Review Board (IRB). During the deployment, we took extensive measures to preserve user privacy and ensure ethical data collection. Each installation of the Raspberry Pis was accompanied by a consent form that clearly outlined the purpose of data collection, the type of data being collected, and how the data would be used. No passive data was collected, and all data was anonymized before it was consumable for analysis.

B State Transition Probabilities for Synthetic Data Generation

For synthetic time series data generation, we model the normal/anomalous state sequence as a 2-state Markov chain with transition matrix

$$P = \begin{pmatrix} 1 - p_{\text{enter}} & p_{\text{enter}} \\ 1 - p_{\text{stay}} & p_{\text{stay}} \end{pmatrix},$$

where state 0 is normal and state 1 is anomalous. We obtain p_{stay} and p_{enter} from the anomaly rate parameter ρ (fraction of anomalous time windows) and the average anomaly duration L (average number of time windows). Once the chain enters the anomalous state, the total number of consecutive anomalous windows follows a geometric distribution with parameter $1 - p_{\text{stay}}$. The expected value of this distribution is given by

$$\mathbb{E}[\text{number of consecutive anomalous windows}] = \frac{1}{1 - p_{\text{stay}}}.$$

Setting this equal to the desired average anomaly duration L gives

$$\frac{1}{1 - p_{\text{stay}}} = L \quad \Longrightarrow \quad p_{\text{stay}} = 1 - \frac{1}{L}.$$

Now, let π_0 and $\pi_1 = \rho$ denote the stationary probabilities of the normal and anomalous states, respectively. The stationary distribution satisfies $\boldsymbol{\pi} P = \boldsymbol{\pi}$, which gives:

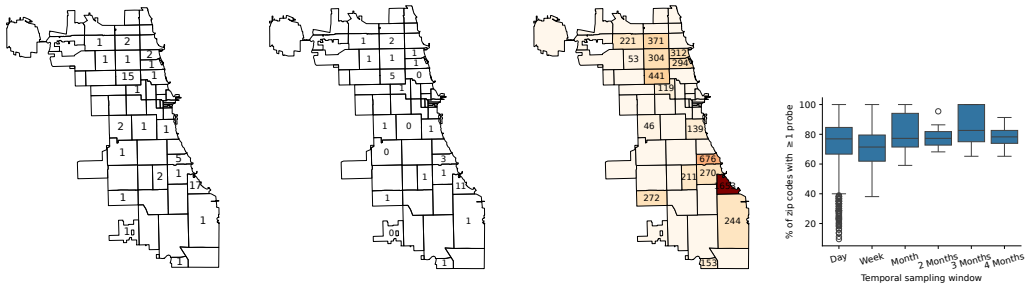
$$\pi_0 \, p_{\text{enter}} = \pi_1 \, (1 - p_{\text{stay}}).$$

Substituting $\pi_0 = 1 - \rho$, $\pi_1 = \rho$, and $1 - p_{\text{stay}} = \frac{1}{L}$:

$$(1 - \rho) \, p_{\text{enter}} = \rho \cdot \frac{1}{L} \quad \Longrightarrow \quad p_{\text{enter}} = \frac{\rho}{L(1 - \rho)}.$$

C Impact of Signal-to-Noise Parameter on Anomaly Detection

To evaluate the impact of the parameter SNR_{min} on the anomaly detection performance, we run an experiment on synthetic data. While varying this parameter between 1.0 and 10.0, we measure the F1-score of detected anomalies against the best performing Jitterbug variant, JB-PELT. Figure 10



(a) Original probe counts. (b) Reduced probe counts. (c) Number of unique anomalies covered by the reduced set. (d) Impact of temporal sampling window.

Fig. 11. Comparison of Comcast probe distribution by zip code in the original and reduced probe sets for the city of Chicago. 17 of 21 zip codes retain at least one probe in the reduced set (11a and 11b). A southern zip code contributes the most number of probes and unique anomalies (11c) from the reduced set. A high fraction of zip codes retain at least one probe across temporal splits of the dataset (11d).

shows the precision, recall and F1-score of detected anomalies for different values of $SNR_{\min}$. We observe that the recall of detected anomalies increases between an $SNR_{\min}$ of 2.0 and 3.0, while the precision decreases for our extension. This is because as $SNR_{\min}$ increases, more anomalies are detected, including subtle jumps that may not be true anomalies, leading to a drop in precision. JB-PELT detects fewer anomalies overall, resulting in lower precision and recall than our extension. The F1-score increases with increasing $SNR_{\min}$ and stays relatively stable between 5.0 and 10.0.

These results suggest that while our extension shows improved performance over JB-PELT across all values of $SNR_{\min}$, it prioritizes detection of prominent anomalies over subtle ones, which may also have a higher impact score. While an overall F1 score of close to 0.5 may seem low, it is important to recognize the class imbalance in our synthetic dataset, where the number of anomalous time windows is much smaller than the number of normal time windows. In this context, an F1 score of 0.5 indicates a significant improvement over random guessing, which would yield an F1 score close to the anomaly ratio ρ (varies between 0.05 and 0.5 in our experiments) due to the induced imbalance.

D Reasoning About Geographic Distribution of Probes

For a network stakeholder designing a probe deployment, one important question may be the following: *with what geographic granularity should probes be placed to achieve a desired coverage of anomalies?* To attempt to answer this question, we analyze the geographic distribution of probes selected by our approach relative to the original probe distribution. We begin by comparing the overall probe counts by zip codes in the original and reduced probe sets. We initially pick zip codes as our default geographic granularity because these are the immediate smaller geographic granularity that appears in IP geolocation databases (e.g., MaxMind [59], IPinfo [1]) after city-level geolocation. The zip code granularity further makes our findings more applicable to a wide range of measurement campaigns that may not have access to more fine-grained geolocation data.

Figure 11 shows a comparison of the original and reduced probe distributions by zip code for the city of Chicago[3]. We calculate the reduced set by continuing with our choice of δ_{IoU} as 0.9, and using all probes that were active throughout our deployment. We then calculate the total number of probes required to cover 95% of anomalies. The reduced set results in a total of 33 probes, one of which is located in a previously unmonitored zip code. A total of 21 zip codes are shown in Figure 11a,

[3]Two probes are intentionally omitted due to being located outside city boundaries.

out of which 17 zip codes (80.9% of total) retain at least one probe in the reduced set (Figure 11b). Our greedy heuristic appears to implicitly preserve the geographic diversity of the probes from the original set, as suggested by the majority of zip codes retaining at least one probe in the reduced set. This further indicates that a geographically diverse probe deployment is important to achieve a high coverage of anomalies for one ISP, *even* when all probes are located within the same city. Finally, we also observe that a southern zip code (60649) contributes the most number of probes (11) and unique anomalies (1653) from the reduced set. This zip code overlaps with the South Shore neighborhood which is known to be racially segregated [83], suggesting that probes deployed in socioeconomically disadvantaged areas may experience greater heterogeneity in network performance that is not captured by probes elsewhere.

To validate that this observation is not an artifact of the underlying data, we also run our greedy heuristic on time-based splits of our dataset using the same parameters as above ($\delta_{IoU} = 0.9$, $c = 1.0$, towards Chicago). To this end, we resample our dataset into daily, weekly, monthly, bi-monthly and quarterly bins and run our heuristic on sliding windows of these bins. For each window, we calculate the fraction of zip codes that preserve at least one probe in the reduced set. Figure 11d shows a box plot of this fraction versus the sampling window size. We observe a high variation in this metric across daily and weekly windows, which is expected due to smaller sample sizes and churn in the probe population. However, for sampling window sizes of a month or more we observe a median fraction of > 0.77 with low variation, suggesting significant robustness of our observation to the underlying data.

We also look at additional geographic granularity choices including Chicago's census tracts and neighborhood boundaries. For the complete dataset, we estimate the *fraction of geographic regions that retain at least one probe* in the reduced set for each choice of geographic granularity. We estimate this metric to be 77.64% ± 7.44% for zip codes, 4.76% ± 0.00% for neighborhoods, and only 2.32% ± 0.14% for census tracts, where the ± values represent one standard deviation. Zip codes, being the larger geographic regions, have a high fraction of regions retaining at least one probe. The low fractions for neighborhoods and census tracts suggest that measurement sampling designers should first consider placing multiple probes at a granularity as large as zip codes before expanding to finer granularities. From Figure 11, we also observe that the two zip codes with 15 and 17 probes in the original set retain 5 and 11 probes respectively in the reduced set. This suggests that some zip codes may require more probes than others to achieve a satisfactory level of anomaly coverage. Prior to actual probe deployments, crowdsourced datasets such as M-Lab [61] or Ookla [66] could be used to identify such areas based on the testing density of users as well as the variability in network performance experienced by these users. Given these insights, we believe that a future research direction could be to combine these data sources with our approach for arriving at more informed probe placement strategies.

Received December 2025; accepted April 2026