Characterizing Home Network Performance Problems

Srikanth Sundaresan, Nick Feamster, Renata Teixeira, Yan Grunenberger, Dina Papagiannaki, Dave Levin

Characterizing Home Network Performance Problems

Srikanth Sundaresan*, Nick Feamster*, Renata Teixeira∓, Yan Grunenberger†, Dina Papagiannaki†, Dave Levin‡
* Georgia Tech ∓ LIP6 † Telefonica I+D ‡ University of Maryland

ABSTRACT

We design, develop, validate, and deploy *WTF (Where's The Fault?)*, a system that determines whether a performance problem in a home network lies with the ISP or inside the home network. WTF uses four independent maximum likelihood detectors to detect both access link bottlenecks and wireless network pathologies with high detection rates and low false positive rates; we use extensive controlled experiments to determine the appropriate thresholds for each parameter that we measure. We implemented WTF as custom firmware that runs in an off-the-shelf home router and deployed it in 64 home networks across 15 countries. The real-world deployment sheds light on common pathologies that occur in home networks. We find that wireless bottlenecks are significantly more common than access link bottlenecks, that the 5 GHz spectrum consistently outperforms the 2.4 GHz spectrum, that many homes experience high TCP round-trip latencies between wireless clients and the access point, and that performance can vary dramatically across wireless devices, even within a single home network.

1. INTRODUCTION

Many people depend on a home network that performs well, yet many factors inside and outside the home can contribute to poor performance in home networks. *Inside the home*, poor placement of an access point, cross traffic from devices within the network, or interference from nearby hosts can result in a bad wireless connection. *Outside the home*, a congested access link, routing problems, poor interdomain connectivity, and many other factors can degrade performance. Unfortunately, neither users nor ISPs currently have a reliable means of determining whether a performance problem lies within the home network or with the access ISP. The ambiguity is both frustrating and costly: Our discussions with several large access ISPs reveal that service calls are costly, ranging from $9–25 per call, and as many as 75% of service calls from customers are usually caused by problems that have nothing to do with the ISP.

In this paper, we develop an algorithm and tool that determines whether network performance problems lie inside or outside the home network (or, in both places). Our tool, *WTF (Where's The Fault?)*, detects access link bottlenecks and wireless pathologies in a home network. WTF comprises four maximum likelihood detectors: one detects access link bottlenecks and the other three detect wireless network pathologies. Together, these detectors allow us to infer properties of the network and the most likely location of performance problems. We base these detectors on network properties that can be easily measured from resource-constrained home routers, which allows us both to design an accurate tool and to implement a longitudinal measurement study. Although WTF does not determine *why* a particular bottleneck or problem exists (*e.g.*, it cannot determine whether a wireless problem results from poor device placement, non-WiFi interference, or other causes), it takes an important first step in helping users and ISPs determine *where* the problem exists, at least to the granularity of whether the problem is inside or outside the home. WTF runs on home access points, where it can directly and continually observe the characteristics of both the access link and the home wireless network.

Our desire to continuously measure the performance characteristics of real home networks for real home network traffic made designing WTF interesting and challenging. To deploy WTF in as many homes as possible, we implemented it as custom firmware that runs on a commodity home router. Although this approach allows us to collect measurements on a low-cost device that users are familiar with (and hence, more than willing and able to install), it introduces a unique set of challenges because the device is so resource constrained. This environment makes it difficult to apply existing bottleneck detection and wireless analysis tools, since they typically require additional affordances (*e.g.*, multiple wireless vantage points, significant trace collection). WTF performs lightweight passive measurement, pre-processes the data, and uploads a concise set of summary statistics to a server, which performs maximum likelihood detection for a variety of conditions, based on several parameters that are easy to collect on the router. To determine the thresholds for our maximum likelihood detectors that achieve high detection rates and low false positive rates, we performed extensive controlled experiments.

We also deployed WTF in 64 homes in 15 countries and measured the extent of wireless and access network performance problems that users experience in these networks; we report on a period covering one month in 2013. Our study yields some interesting findings: First, most homes in our deployment have wireless problems most of the time. Second, the 5 GHz wireless band consistently outperforms the 2.4 GHz band, likely because it has less contention and interference. Third, TCP round-trip latencies between a home wireless access point and devices in the home can be high; in many cases, the round-trip latency introduced by the wireless network is a significant fraction of the end-to-end round-trip latency. Finally, performance varies across devices, even

within a single home.

We offer two important contributions: (1) the design, development, and validation of WTF, a tool that both accurately detects home access link and wireless network pathologies and is lightweight enough to run continually on a home router; (2) a detailed characterization of the nature and extent of performance problems that commonly arise in many home networks. The Federal Communications Commission is planning a wider deployment of WTF, and we plan to release WTF to the community by the end of 2013. Our results lend insight into home networks that we believe have potentially important ramifications for ISPs, content providers, and users. In particular, our results suggest that it is worth spending effort to improve home wireless network performance, in addition to the extensive attempts to optimize latency in other parts of the network and end hosts.

2. RELATED WORK

Measuring and diagnosing network performance issues has a long history that has spanned many types of networks. In this section, we briefly survey prior approaches and discuss why home networks require a new approach.

There have been many previous approaches to diagnosing wireless networks. One approach is to deploy passive traffic monitors throughout the network. Kanuparthy *et al.* [17] develop a tool to detect common wireless pathologies (such as low signal-to-noise ratio, congestion, and hidden terminals) by using both active probes and an additional passive monitor deployed within the network. Cooperative techniques can also diagnose certain classes of problems like hidden terminals and conflict graphs [4, 22]. Pervasive monitoring approaches work well in enterprise networks [2, 8, 24]: Mahajan *et al.* study the wireless performance in a large network by collecting traces from many vantage points and piecing them together [21]. Judd *et al.* [14] characterize the link-layer performance of 802.11 under various different cases such as clear channels and with hidden and exposed terminals. Unfortunately, it is difficult to perform this kind of extensive monitoring in many home networks, since it requires deploying equipment beyond that which a normal user is typically willing to install or have installed in their home.

Other approaches have monitored wireless networks with custom hardware [7, 20, 24–26]. RFDump [20] is a tool built on GNU Radio and USRP to monitor heterogeneous wireless networks with devices such as Bluetooth. AirShark [25] exploits a recent 802.11 chipset to collect spectrum samples, allowing for detection of non-WiFi interference. In contrast, WTF runs on off-the-shelf access points; this approach allows us both to achieve widespread deployment and to leverage the existing BISmark infrastructure [30].

Several techniques for detecting bottlenecks in wide-area networks exist; these approaches typically rely on active measurements [5, 11–13, 16, 19, 27–29]. PathNeck [11, 12], for instance, is an active probing tool which can accurately locate bottleneck links in a wide-area network. Unfortunately, in home networks, active techniques have two key disadvantages: they may not accurately reflect the actual performance users experience (and even interfere with it), and additional cross-traffic can actually affect the wireless network's performance. Thus, we design a passive monitoring technique for bottleneck detection in WTF.

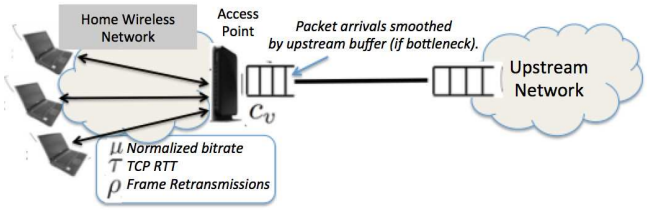


Figure 1: *WTF runs on the access point between the home network and the access link, thus offering a unique vantage point for observing pathologies on either side.*

WTF draws inspiration from several previous diagnosis techniques. Zhang *et al.* develop T-RAT [32] to analyze TCP performance. T-RAT estimates TCP parameters such as maximum segment size, round-trip time, and loss to understand flow behavior. Katabi *et al.* [18], use entropy in packet interarrival time to estimate shared bottlenecks. Biaz *et al.* [6] use packet interarrival times for distinguishing between different kinds of losses. WTF is similar to some of the approaches used in these papers (*e.g.*, it uses packet interarrival time as input to a maximum likelihood detector for access link bottlenecks), but we tailor our approach so that it only relies on data that can be easily collected from a home router.

Home networks can also be subject to performance problems caused by explicit policy or configuration decisions. Netprints [3] is a diagnostic tool for home networks solves problems arising due to misconfigurations of home network devices including routers. Other work has explored the extent of performance degradations due to service discrimination [9, 10, 15, 31]. WTF focuses on performance problems that arise from limited bandwidth, high retransmission rates, and so on. Incorporating service discrimination into a broader explanatory model of *why* these performance problems exist is an important area of future work.

3. DETECTION ALGORITHM

We develop methods to attribute performance problems to either the home network or elsewhere and to provide additional details about the reasons for the pathology when possible. WTF aims to detect two distinct scenarios: (1) the access link is the bottleneck, and (2) the wireless link is the bottleneck. We describe the problem setup, the intuition behind WTF's detection algorithms, and assumptions and limitations of our approach. We then describe the controlled experiments that we use to design maximum likelihood estimators to detect these scenarios.

Parameter	Description
Access Link Bottleneck (B)	
c_v	Coefficient of variation of interpacket arrival time
Wireless Pathology (W)	
μ	Average wireless bitrate of frames, normalized by max. bitrate supported
τ	TCP RTT between the AP and the client
ρ	Frame retransmission rates

Table 1: *The random variables that WTF measures and the roles that they play in helping localize faults to either the home network or the access link. For each random variable that we observe and measure, we design a maximum likelihood estimator to detect whether or not the pathology exists.*

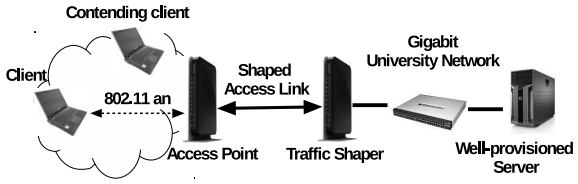


Figure 2: *Controlled experiment setup.*

3.1 Approach

Figure 1 shows how a typical home network connects to the wide-area Internet, and the metrics that WTF collects, which are also described in Table 1.

Overview of detection. WTF collects its measurements from a single access point in the home network. Although this approach has many limitations (*e.g.*, a single vantage point prevents identifying certain classes of problems such as hidden terminals), it also has many advantages, such as being able to monitor real user traffic at multiple layers at the boundary between the home network and the rest of the wide-area Internet. The approach also facilitates a larger deployment in homes, since no specialized hardware is required. We identify a collection of features that can be easily measured from the access point and also provide good indications of the quality of both the wireless network and the access network. WTF performs two independent detections:

- **Determine whether the access link is a bottleneck.** WTF determines whether the access link is bottlenecked by computing the coefficient of variation of packet interarrival time, c_v, and comparing it against a threshold. (Section 3.3.1).
- **Look for pathologies in the wireless network.** WTF analyzes 802.11 frames for different parameters for estimating the quality of the wireless link: μ, the normalized bitrates; and ρ, the retransmission rates. From TCP traces, it also computes τ, the TCP round-trip latency between the access point and the client. (Section 3.3.2).

Selecting detection thresholds: Maximum likelihood estimation. For each parameter that we evaluate, we design a maximum likelihood detector that treats the observed values of the parameter as a random variable to determine whether it is more likely or not that the pathology has occurred.

For example, to determine whether the access link is the bottleneck, we calculate c_v, the coefficient of variation (the standard deviation divided by the mean) of packet interarrival times on the WAN side of the access point. Our detector is based on a decision rule that determines whether the "access link bottleneck" event, B, occurs given a particular observed value of c_v during a particular time period. We first compute the conditional probabilities $f(c_v|B)$ and $f(c_v|\overline{B})$ in our controlled setting, where we use our ability to control the throughput of the upstream link to introduce a bottleneck on the access link. We then define our decision rule in terms of the likelihood ratio:

$$\Lambda = \frac{f(c_v = v|B)}{f(c_v = v|\overline{B})}$$

where c_v is the coefficient of variation of packet interarrival time for packets over the observation window. When Λ is greater than some threshold γ, the detector says that the access link is the bottleneck (*i.e.*, it is more likely than not, given the observation of c_v, that the prior is the event B). We can tune the detector by varying the value of the detection threshold, γ; higher values will result in higher detection rates, but also higher false positive rates. Given Λ, we can thus determine the probabilities of a false positive and detection for different values of γ.

These ranges of false positives and detection are commonly known as a receiver operating characteristic (ROC) for a decision rule. We develop a maximum likelihood detector for each parameter (c_v, μ, ρ, and τ) as detectors for the corresponding pathologies B and W, as outlined in Table 1.

3.2 Controlled Experiments

We built a testbed to run controlled experiments to evaluate each of WTF's detectors; Figure 2 shows this testbed. The testbed has an access point, its LAN, a network shaper upstream of the access point, a well provisioned university network, and servers in the university network. The access point is a Netgear WNDR3800 router running OpenWrt. To change the downstream throughput of the emulated access link, we use `tc` and `netem` on a second WNDR3800 router running OpenWrt. We run our throughput tests against servers in the same well provisioned university network to avoid potential wide-area effects.

We use this testbed to explore WTF's behavior for a variety of scenarios. For each maximum likelihood estimator we select an appropriate threshold based on its receiver operating characteristic (ROC) that yields a high detection rate and a low false positive rate. We run two sets of experiments using the testbed, for the two pathologies we are trying to detect. For the access link bottleneck scenario, we use the traffic shaper to shape the link to different throughput levels while keeping the wireless link constant. In this case,

identifying the ground truth is straightforward, as we know the capacities of both the wireless link and the shaped access link. We use 802.11a for the wireless link with the capacity at 21 Mbps over TCP. We generate over 1200 samples with 11 different emulated access link throughputs varying from 3 Mbps to 100 Mbps.

For the wireless pathologies, introducing pathological cases and determining ground truth is more difficult. Rather than directly controlling wireless throughput, we must directly subject the network to certain conditions and then observe the achieved TCP throughput. We label the wireless as pathological (event W) if the achieved throughput is less than 50% of the capacity of the channel (which we determine by running tests under a "clean" environment). To introduce wireless pathologies, we run two sets of experiments: (1) reduce capacity by degrading channel quality: we do this by positioning the host at different distances from the access point, and with multiple obstructions, and also transient problems by human activity. (2) reduce the available capacity of the channel by creating contention with another host that sends constant UDP traffic, with the first host close to the access point.

For each experiment, we run a TCP throughput test using `iperf`. To minimize interference that we do not introduce ourselves, we use the 5 GHz spectrum, which is less congested than the 2.4 GHz range in our testbed. In our repeated controlled experiments, we found that the wireless channel in our testbed delivers a TCP throughput of about 82 Mbps on 802.11n. If the wireless throughput drops below 40 Mbps as a result of the conditions that we introduce, we label the corresponding condition as a wireless pathology (event W). We then use apply maximum likelihood detection to the random variables that we describe in Table 1 to determine the most effective thresholds for detecting these wireless pathologies. We generate over 1,500 samples over many wireless operating conditions, with the TCP throughput varying from less than 10 Mbps to more than 70 Mbps using these techniques.

3.3 Detector Design and Validation

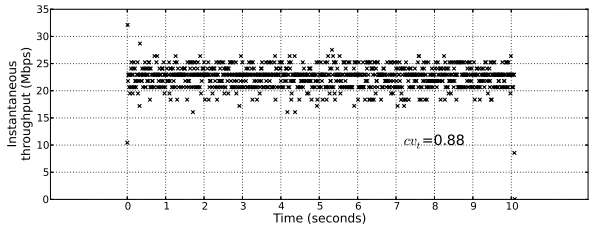
We now describe how we use different parameters that we can measure from the home access point to detect different pathologies in the home network.

3.3.1 Access link bottleneck

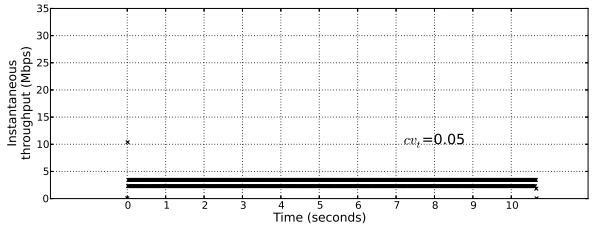
We now describe how we use the coefficient of variation of packet interarrival times to detect access link bottlenecks.

Intuition: Bottlenecks smooth packet arrival rates. Because a bottleneck link services packets at a rate slower than they arrive, queues build up at the link, and the link thus paces packets at a relatively even rate. Packets upstream of the bottleneck will arrive according to the natural variation induced by TCP congestion control, but downstream of the bottleneck link, packets will be more evenly spaced. We assume that the most likely bottleneck upstream of the home network is the access link, so *all* flows are buffered, which allows us to use the overall packet distribution for detection.

We expect to see high variance in packet interarrival times before the bottleneck link due to congestion control, but significantly lower variance after the bottleneck link itself because the buffer smoothes packet arrivals. Figure 3 shows this effect: It shows the instantaneous TCP throughput at a granularity of 10 ms, as measured from the access point. In Figure 3a, the access link throughput is 100 Mbps; the



(a) Access link is not the bottleneck. *Instantaneous throughput at the WAN interface varies at short time scales due to high variance in packet inter-arrival times.*



(b) Access link is the bottleneck. *Instantaneous throughput at the WAN interface is steady, due relatively uniform packet interarrival times caused by upstream shaping.*

Figure 3: *Behavior of packet inter-arrival times.*

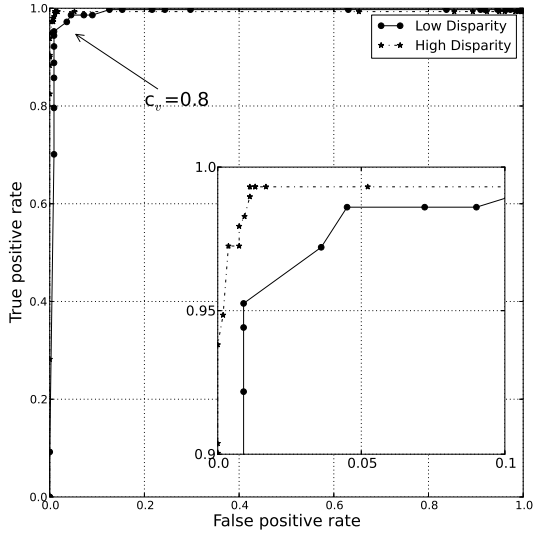


Figure 4: *Receiver operating characteristic for access link bottleneck detection using the coefficient of variation of packet interarrival time. "Low disparity" indicates conditions when the access link throughput is between 18–22 Mbps, close to the throughput of the wireless network.*

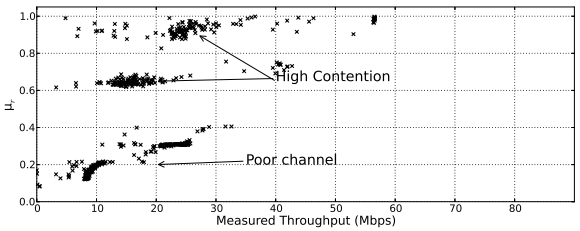


Figure 5: Normalized average bitrate. *Lower values of μ indicate a poor channel.*

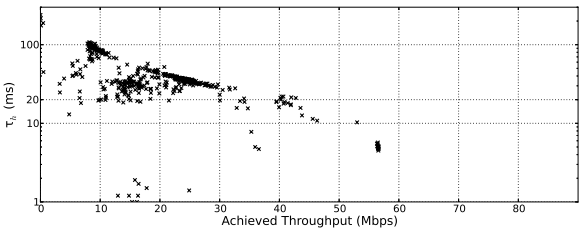


Figure 6: TCP RTT between client and access point. *τ decreases as throughput increases and the disparity between the access link and the wireless link throughput decrease.*

wireless link is the bottleneck because the maximum TCP throughput it can support is about 21 Mbps. In Figure 3b, we shape the access link to 3 Mbps, significantly lower than the wireless capacity. In this case, throughput is less variable. Indeed, the coefficient of variation for packet interarrival times, c_v, when the access link is the bottleneck for this example is 0.05; in contrast, when it is *not* the bottleneck, c_v is 0.88.

Choosing a threshold for c_v. Based on observed c_v, we can determine whether it is more likely or not that the access link is the bottleneck. We develop a maximum likelihood detector based on the two different conditional probability distributions, $f(c_v|B)$ and $f(c_v|\overline{B})$ to determine the threshold. We first evaluate the detection accuracy of the algorithm for different values of the detection threshold for c_v. Figure 4 shows the receiver operating characteristic for this detector. When the threshold is low (close to zero), it will always identify the access link as not the bottleneck, and when it is high (close to one), it will always identify the access link as the bottleneck. We test detection over for access link throughputs from 3 Mbps to 100 Mbps keeping the wireless throughput at 21 Mbps. Our results indicate that detection accuracy remains high for a wide range of threshold settings, particularly between 0.4 and 0.85. We use a threshold of $c_v < 0.8$ to declare the access link the bottleneck.

The inset in Figure 4 splits the results of this experiment, into two scenarios: *low disparity*, indicating cases where we shaped the access link throughput to rates between 18–22 Mbps (*i.e.*, to a rate that was very close to the wireless throughput); and *high disparity*, indicating all other cases. The ROC shows that the detector is slightly less accurate when the throughputs of the access link and the wireless link are closer, but even in these cases the detector is still very accurate, achieving nearly a 95% detection rate for less than a 2% false positive rate.

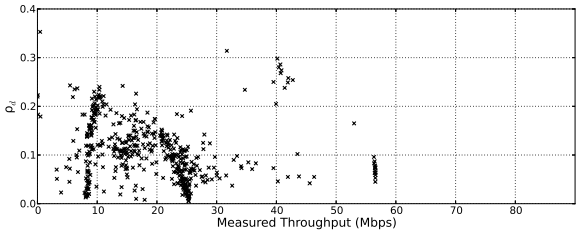


Figure 7: Frame retransmission rate. *Higher rates indicate poor channel.*

3.3.2 Wireless pathologies

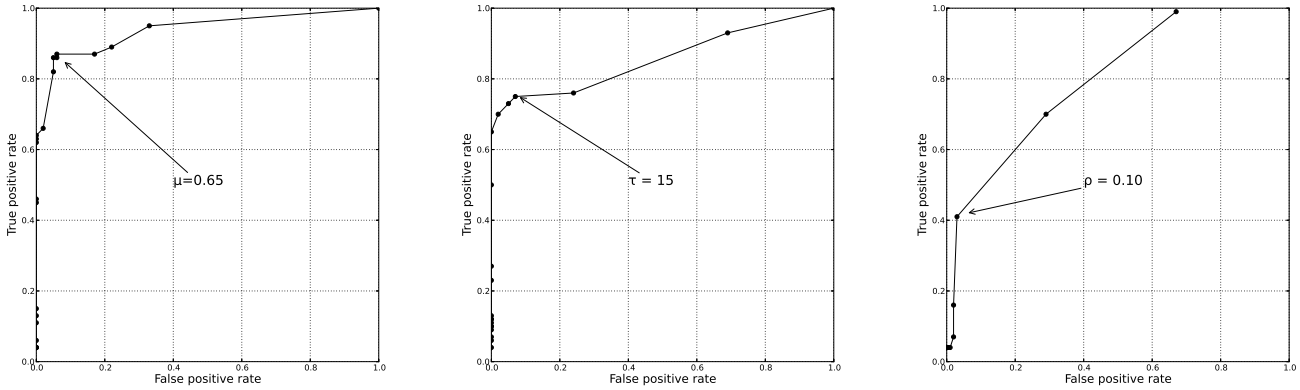
Wireless pathologies in home networks include poor channel quality, a lossy channel, and contention. We develop three detectors for determining wireless network quality:

- *Normalized average wireless bitrate (μ).* We use wireless bitrate, normalized by the maximum bitrate supported by the 802.11 variant, to detect poor wireless channel quality.
- *TCP round-trip time to clients (τ).* We use the TCP round-trip time between the access point and the client to detect significant bottlenecks on the wireless channel, caused potentially due to contention in the channel.
- *Retransmission rate (ρ).* We use frame retransmission rates to detect a lossy wireless channel.

These detectors determine whether the wireless network is experiencing certain pathologies (*e.g.*, loss). We do *not* aim to determine the underlying cause for these pathologies (*e.g.*, interference, poor signal strength); we leave the questions of root cause for future work.

Normalized average wireless bitrate. IEEE 802.11 bitrate adaptation techniques adjust the transmission bitrate as wireless channel conditions change. Although these techniques usually adapt rates even under benign conditions to determine the channel quality, rate adaptation is typically more frequent when the channel quality is poor, as wireless senders typically reduce the bitrate in response to poor channel quality. Thus, we can use the average wireless bitrate, normalized by the maximum bitrate supported by that channel, μ, as an indicator of a poor wireless channel. Figure 5 shows that μ tends to be low when the wireless channel quality is poor, (when the achievable throughput is less than 40 Mbps) and high when the channel is good. When we introduced contention (with good channel quality) throughput is low, but bitrates are high; this means that normalized bitrate is a good detector for poor or lossy channel, but not a good detector when the throughput is constrained for other reasons.

As with the access link bottleneck detector, we designed a maximum likelihood estimator based on the conditional probability distributions, $f(\mu|W)$ and $f(\mu|\overline{W})$, as previously described. We then varied the detection threshold and evaluated the detector under different wireless conditions. Fig-



(a) *Normalized wireless bitrate.* **(b)** *TCP RTT between AP and client.* **(c)** *Frame retransmission rate.*

Figure 8: *The ROCs for individual parameters that WTF uses to detect different wireless network pathologies.*

ure 8a shows the corresponding ROC curve. We flag the wireless channel as poor when $\mu < 0.65$. This threshold yields a detection rate of about 85% for a false positive rate of 7%. This detector assumes that the wireless clients perform bitrate adaptation in response to retransmissions, which is the case with most wireless clients—in the case of our access point, the bitrate adaptation is the Minstrel bitrate adaptation algorithm used by the `ath9k` driver.

TCP round-trip time from access point to client. Figure 5 showed that throughput can be low even if wireless bitrates are high and frame retransmission rates are low—most likely due to wireless channel contention, or simply because the wireless channel capacity is not sufficient. Because the available capacity of 802.11n is quite high (more than 80 Mbps over TCP in our experiments), it is likely that such wireless bottlenecks are caused due to contention. Thus, we use the TCP round-trip time between the client and the access point, τ, to detect cases where the wireless channel quality is good, but the channel is still constrained. Figure 6 shows how the local network RTT (between the access point and the client) varies as a function of achieved throughput. The figure shows that high TCP round-trip latency between the client and the access point generally correlates with lower achieved throughput. When there is no bottleneck in the wireless, the round trip time between the client and the access point should be on the order of a few milliseconds. On the other hand, in the presence of contention (or any case where the access link throughput is much greater than the wireless throughput), buffering and backoff can introduce delays that appear as high TCP round-trip latencies. We designed a maximum likelihood detector based on the distributions $f(\tau|W)$ and $f(\tau|\overline{W})$; Figure 8b shows the corresponding ROC, where a threshold of $\tau > 15$ ms yields a detection rate of 75% and a false positive rate of 7%.

Frame retransmission rate. When bitrate adaptation does not have enough time to adjust the bitrate (due to varying channel conditions), the normalized bitrate might not be a good enough detector. In such cases retransmission rates are still high. Figure 7 shows the relationship between measured throughput and retransmission rate. As expected, there is some negative correlation between frame retransmission rate and measured throughput, although the correlation is not as strong as it is with other variables. We designed a maximum likelihood detector based on the distributions $f(\rho|W)$ and $f(\rho|\overline{W})$; Figure 8c shows the corresponding ROC, where a threshold of $\rho < 0.1$ yields a detection rate of 40% and a false positive rate of 1%.

3.3.3 Putting it together

Our results indicate that the best detector of wireless pathologies is the normalized bitrate, likely because any problems in the wireless channel are typically reflected by an adaptation to a lower bitrate. We also see that none of the three detectors is perfect, but they complement each other well and are useful in different scenarios. For example, the bitrate detector works when the wireless pathology is persistent, thus allowing time for bitrate adaptation to find a lower rate. On the other hand, high retransmission rates can serve as a useful indicator for transient pathologies where the wireless bitrate adaptation has not yet adapted (or will not adapt), but it is less useful for persistent problems, where bitrate adaptation may have adjusted to keep retransmission rates low. TCP round-trip times can serve as a good detector in general cases where the wireless is a bottleneck, including those where the bottleneck is due to other reasons than a poor channel, such as contention. It does not work as well when the buffering is sporadic, as might be the case when the channel quality is continually changing, although other parameters can work well in those cases.

Our goal is to detect event W (the wireless is pathological), so we can therefore use the three detectors together to

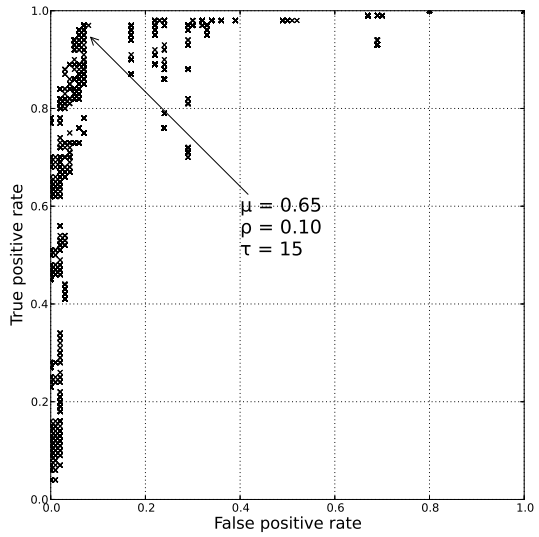


Figure 9: *ROC for the detector that uses all three wireless variables as detection parameters. The ROC is not continuous, since each point reflects a particular set of thresholds for (μ, ρ, and τ).*

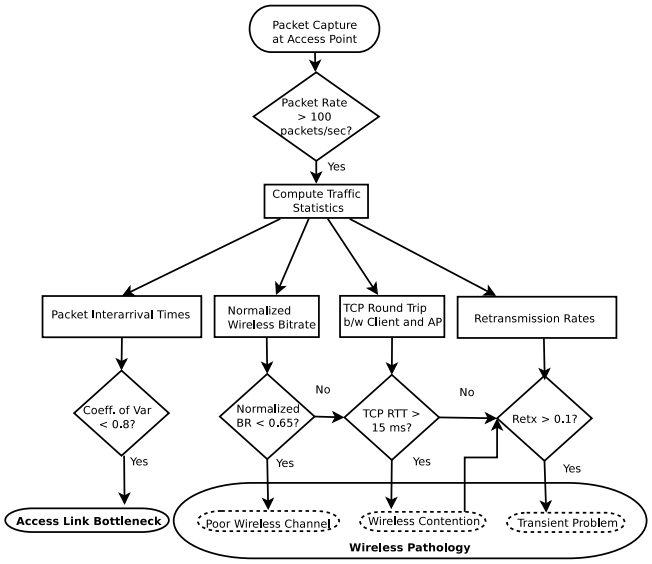


Figure 10: *Combining each of the detectors to create a single combined detection algorithm for access link bottlenecks (event B) and wireless pathologies (event W).*

construct a more complete picture of wireless pathologies, as shown in Figure 10. The detectors themselves can give us a better idea about the causes, but we do not validate that in this paper, but we explain why it makes sense. Our approach results in a 97% detection rate for a 7% false positive rate over 1,500 samples for all wireless pathologies, as shown in the ROC curve in Figure 9; operators who wish to achieve lower false positive rates can, of course, select thresholds with lower detection rates. We also saw a wide range of values for all three parameters that give us high detection rates; which indicates that the detection mechanism is highly robust and not overly sensitive to choice of thresholds.

4. SYSTEM DESIGN AND DEPLOYMENT

We describe a prototype of WTF that we have deployed in 64 homes. Running on commodity access points in so many homes posed several technical challenges: Although commodity routers offer a low cost and familiar form factor, they have limited computation and storage capacity. Moreover, keeping users engaged requires that WTF be unobtrusive and respect user privacy. We detail the data WTF collects and how the overall system design addresses these challenges.

4.1 Measurements

WTF uses passive measurements, which (unlike active measurements) do not risk introducing contention that could affect the very conditions and performance that we seek to characterize. Further, passive measurements more accurately reflect the actual performance that users experience. To facilitate deployment across a large number of homes WTF collects traces from only a single vantage point; this approach allows us to run WTF within the context of any existing home network, without deploying additional (or customized) hardware. In contrast to several existing wireless diagnostic tools that use multiple monitoring points [8, 21], WTF may have more difficulty diagnosing certain classes of anomalies (*e.g.*, hidden terminals), but it can still detect a useful set of pathologies.

There are many ways to collect the data used in the detection algorithm that we described in Section 3. To facilitate deployment, WTF collects only measurements that were easily accessible from a resource-constrained home router. Additionally, we designed WTF's data collection to be as lightweight and concise as possible, to facilitate fast and unobtrusive uploads to a central analysis server. WTF collects the following measurements:

- `pcap` *traces of connections.* We collect packet traces with `tcpdump` from both the WAN and the wireless interfaces (each router has two). Packet traces from the WAN interface provide information about TCP connections and IP packets flowing through the access point. The wireless interfaces (in monitor mode) capture radiotap headers [23], which, for each frame, include: the source and destination stations, the bitrate used, and whether the frame was retransmitted (but not how many times it was retransmitted). The server computes bitrates and retransmission rates independently for each device.
- *ARP information.* This data provides the device MAC ID-to-IP address mapping for end points in the home network.
- *Connection tracking information from Network Address Translator (NAT) module.* To obtain information about the end point of TCP connections inside the home, we collect a snapshot of the `conntrack` file that maps WAN ports to

LAN IP addresses and ports.

4.2 Design and Implementation

We use Netgear's WNDR3700/3800 platform, which has an Atheros chipset with a 450 MHz processor, one 802.11gn radio, and one 802.11an radio. The 3800 has 128 Mbytes of RAM, and the 3700 has 64 Mbytes of RAM. The devices run OpenWrt, with the ath9k wireless driver. The driver uses the Minstrel rate adaptation algorithm, with the default setting to a maximum bitrate of 130 Mbps.

Due to the resource limitations on the access point, we perform data collection and some amount of limited processing locally but push most processing and analysis to a central server. WTF first processes the WAN `pcap` traces to extract timestamps of arriving packets and information about individual flows such as RTT on either side of the access point, and the number of packets in each connection (using `tcptrace` [1]). Performing the trace at the access point allows us WTF to decompose clearly identify the latencies between the access point and either and each respective endpoint. The access point tracks packets and the corresponding ACKs to compute the RTTs. WTF also processes the radiotap traces to obtain the source and destination MAC addresses and the frame control bits for each frame.

To respect user privacy, WTF anonymizes all IP addresses and MAC addresses completely using SHA-256 and a per-router secret salt as the data is collected on the router. The router discards all private information and uploads the pre-processed to the server, at which point it deletes the local copy of the data. The data is stored in a database where the diagnosis and longitudinal analysis portions of WTF reside. *All aspects of this study have been reviewed and approved by our university institutional review board (IRB).*

WTF considers only the instances where traffic exceeds 100 packets per second, to ensure a reliable computation of c_v. Before computing c_v for an interval, WTF also discards outlier samples for cases where the packet inter-arrival time exceeds the average plus two standard deviations.

Continuous data collection and analysis would impose a significant burden on commodity access points. Apart from CPU intensive tasks such as monitoring traffic on multiple interfaces, the access point must also collect a significant amount of data. To minimize the CPU load and the amount of data uploaded, the current implementation of WTF collects data once every 5 minutes on average for 15 seconds per iteration. Sampling provides insight into the overall nature of each home network and facilitates rapid development and deployment, but it does not allow us to obtain fine-grained characteristics (*e.g.*, conditions that vary with high frequency). We are currently developing a version of WTF that performs continuous monitoring.

5. RESULTS

To understand where performance problems tend to occur in real home networks, we deployed WTF in 64 homes. Table 2 summarizes our deployment and the characteristics of the home networks in this deployment. Our results lead to the following findings, which we highlight in respective subsections: (1) wireless network bottlenecks are common; (2) the 5 GHz wireless band consistently outperforms the 2.4 GHz band; (3) TCP latencies on the wireless network inside a home can be a significant fraction of overall round-trip latency; and (4) the performance of individual devices in the same home network can vary considerably. We now explore each of these results in more detail.

Total # of homes	64
Duration	Mar 6 – Apr 6, 2013
Total # of countries	15
2.4 GHz	
Active devices	163
Devices per home	2.5
5 GHz	
Active devices	63
Devices per home	1

Table 2: *We deployed WTF in 64 households in 15 countries across four continents.*

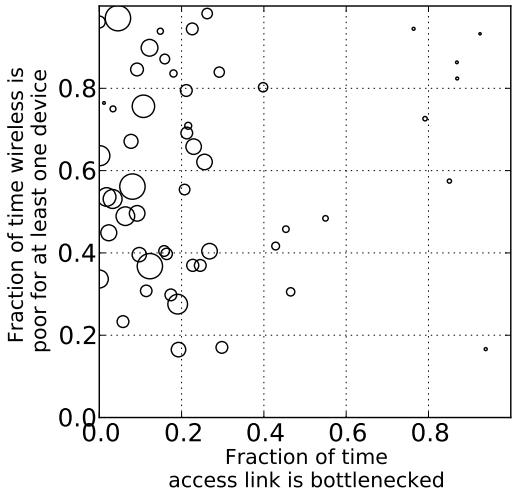


Figure 11: *Prevalence of pathologies home networks. Each circle represents one home (circle area is proportional to downstream throughput). Poor wireless connectivity is much more common than are access link bottlenecks.*

5.1 Wireless Bottlenecks Are Common

We study the relative frequency of the two types of pathologies that our detectors from Section 3 detect, based on the threshold settings that we derive from our controlled experiments. Good wireless performance with low access link utilization suggests a lightly used network (and the possibility of even downgrading the service plan without adverse effects). High utilization and a poor wireless suggests that there are potential wireless bottlenecks that may not be currently affecting performance, yet improving wireless performance might yield significant performance improvements when done in conjunction with an access link upgrade. Figure 11 plots the fraction of time the access link is bottle-

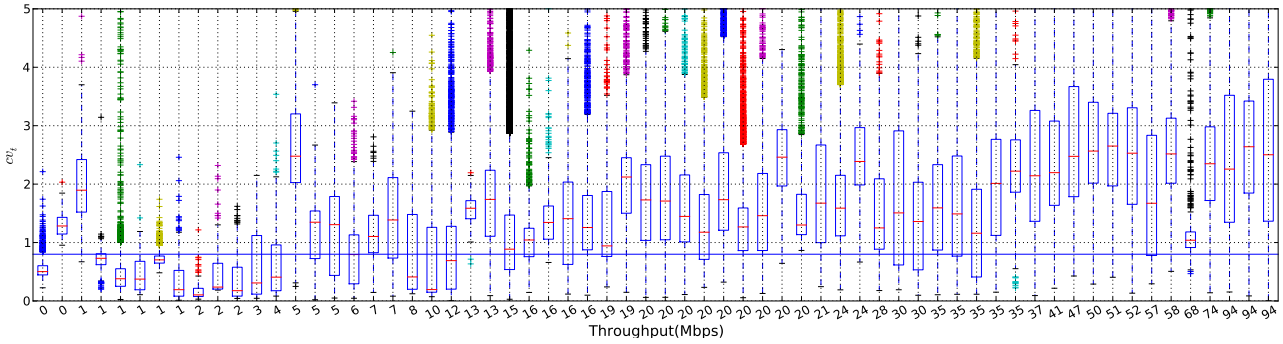


Figure 12: *c_v values for all home networks in our study; values below the horizontal line indicate consistent access link bottlenecks. The horizontal line shows the threshold for c_v of 0.8, below which we declare the access link to be the bottleneck. None of the home networks whose access links have downstream throughput greater 35 Mbps experience a significant access link bottleneck.*

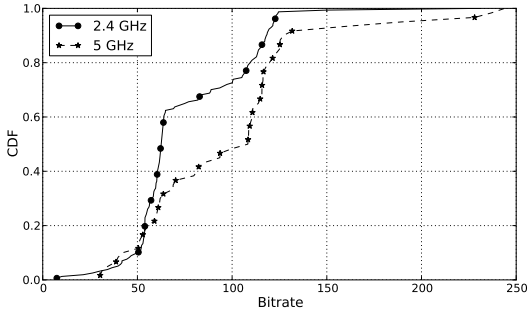


Figure 13: *Distribution of wireless bitrates for devices in both the 2.4 GHz and 5 GHz spectrums, for all devices in the deployment.*

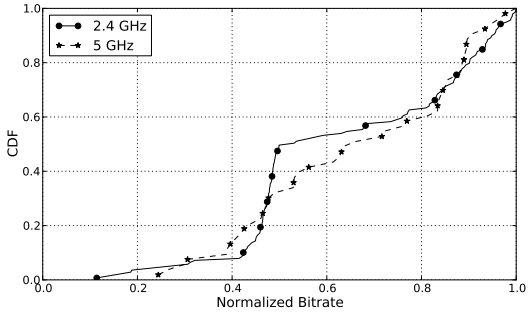


Figure 14: *Distribution of median normalized bitrates, μ, for devices in both the 2.4 GHz and 5 GHz spectrums. Devices do not achieve maximum bitrate, especially in the 2.4 GHz range, and about 50% of the devices experience poor wireless channels at least half of the time.*

necked versus the fraction of time that at least one active wireless device is experiencing a potential bottleneck. Each circle represents a single home network; the area of the circle is proportional to the downstream throughput of the access link for that home. The results show that *most homes in our deployment have wireless problems a significant portion of the time, and are likely bottlenecked by the wireless network*.

We now look at the nature of these access link bottlenecks in more detail. Figure 12 shows the distribution of the coefficient of variation for packet inter-arrival time, c_v, for all 64 homes in our deployment. The box plot shows the interquartile range of c_v when traffic on the access link exceeds 100 packets per second (*i.e.*, when the network is not idle). We observe that *none of the homes with downstream throughput greater than 35 Mbps experience a significant access link bottleneck* (which we define as having the 25th percentile value of c_v falling below the bottleneck detection threshold). We also observe two other features: First, c_v generally increases as access link speed increases. This result makes sense: high downstream throughput reduces the likelihood of the access link being bottlenecked with traffic and increases the likelihood of the wireless being the bottleneck. Second, we observe large variations in c_v, even among access links of similar throughputs. This variation results from the diversity of wireless conditions and usage patterns across households. Home networks with higher access link throughput also tend to have higher c_v values, since it is less likely that the access link is a bottleneck in those cases.

5.2 The 5 GHz Band Performs Better

We analyze the performance that devices in home wireless networks achieve and how performance varies depending on whether devices are on the 2.4 GHz band or the 5 GHz band. Our hypothesis was that devices on the 5 GHz band would perform better because there are generally fewer devices (and surrounding access points) in the 5 GHz band, and that the 5 GHz band also has less non-WiFi interference (*e.g.*, microwaves, baby monitors).

Figure 13 plots the CDF of the median bitrate for all devices in all homes, for both the 2.4 GHz band and the 5 GHz bands. Only 30% of 2.4 GHz devices see median bitrates above 65 Mbps; in contrast, more than 50% of devices in the 5 GHz spectrum see bitrates greater than 100 Mbps, likely because the 5 GHz band is less crowded. Figure 14 shows the median bitrate per device for each home network, normalized by the maximum supported bitrate of the corresponding wireless protocol (by default 130 Mbps but up to 300 Mbps

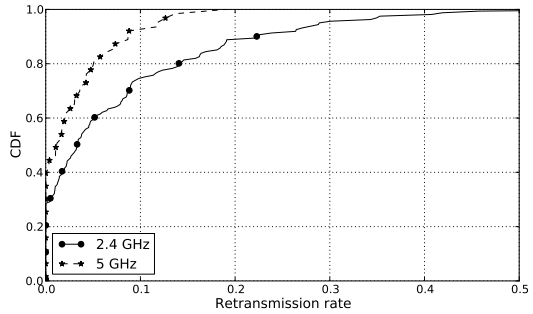


Figure 15: *Distribution of median retransmission rates, ρ, for devices in both the 2.4 GHz and 5 GHz spectrums. Retransmissions are higher in the 2.4 GHz spectrum, where nearly 30% of devices see a median ρ value corresponding to poor wireless channel conditions.*

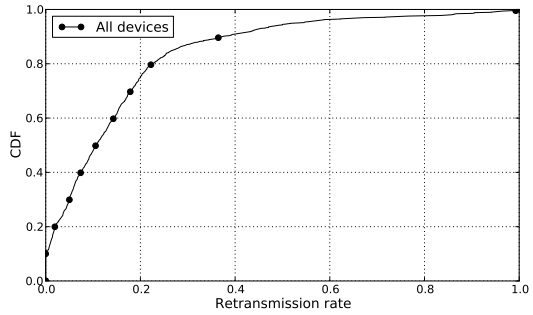
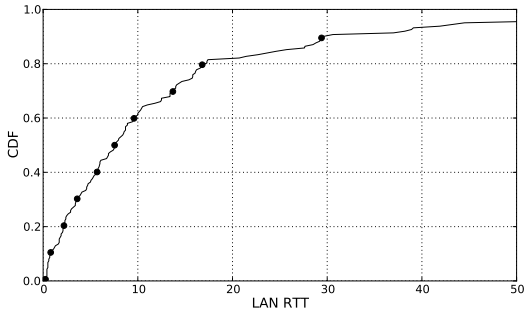
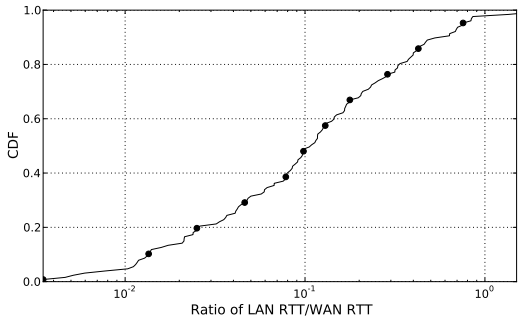


Figure 16: *CDF of ρ for one home network in our testbed; the frame retransmission rates are greater than 0.1 about 50% of the time.*



(a) *Distribution of TCP round-trip time between the access point and client, τ, across all devices in our study. About 30% of all devices experience TCP round-trip latencies corresponding to poor wireless channel conditions ($\tau > 10$ ms).*



(b) *The distribution of the median ratio of the LAN TCP round-trip time to the WAN TCP round-trip time across all flows for that device, across all devices.*

Figure 17: *TCP round-trip latencies between the access point and the client.*

for 802.11n, and 54 Mbps for 802.11a/g).

Our results also show that many devices, especially those in the 2.4 GHz range, often operate well below the maximum bitrate supported by the protocol. Figure 15 shows the retransmission rates for all devices across all homes; the result shows similar trends with respect to the 2.4 GHz and 5 GHz ranges: retransmissions are common in the 2.4 GHz band, with about 20% of devices having retransmission rates above 10%. Figure 16 shows one particular home that experiences frequent wireless bottlenecks. The retransmission rates (from the client to the access point) are more than 0.1 about 50% of the time. We were fortunate to be able to visit this home network to investigate the cause of common pathologies: We observed that the access point is about 5–8 meters away from the location where the occupant typically uses his wireless laptop, and that there were multiple walls in between the access point and the laptop.

5.3 Intra-home Latency Can Be High

The TCP round-trip time between the wireless access point and a wireless client should be on the order of a few milliseconds. In cases where the wireless network becomes a bottleneck, however (*e.g.*, in cases of contention), TCP round-trip latency may increase significantly. We study the extent to which TCP round-trip latency inside a home exceeds 15 milliseconds, the threshold for τ that WTF uses to determine that a home network is experiencing poor wireless performance.

Figure 17a plots the distribution of the median TCP round-trip latency between the device and the access point across all devices in our study. The median device on the local wireless network sees a median latency of about 8 ms, but nearly 25% of the devices experience local TCP round-trip latencies, τ, greater than 15 ms (*i.e.*, a significantly degraded wireless network) more than half of the time. We investigated the performance of these homes further to determine whether they had anything in common, such as having lots of devices (suggesting high contention rates), but we have not yet uncovered any commonalities across these homes.

Because WTF performs its analysis on passive traces, we have the luxury of analyzing the performance of the home network relative to the wide-area network performance for distributions of real user traffic in the homes across our deployment. We use these traces to compare the round-trip times between the devices and the access point to the round-trip times from the access point to the wide-area destination for each flow. We define the *median latency ratio* for a device as the median ratio of the LAN TCP round-trip time to the WAN TCP round-trip time across all flows for that device. Figure 17b shows the distribution of the median latency ratio across all devices. The result shows that 30% of devices

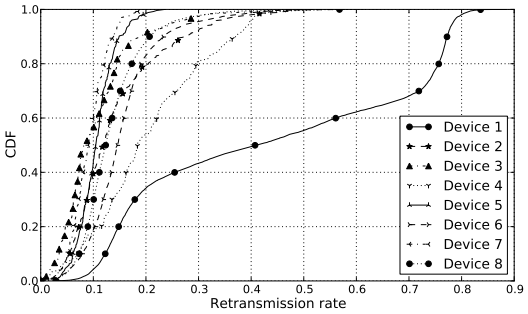


Figure 18: *The retransmission rates between the access point and clients in a single home network. In this home retransmission rates are high. Interestingly, one device has a significantly higher retransmission rate.*

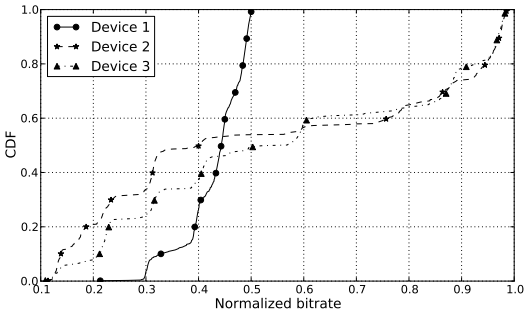


Figure 19: *CDF of bitrates for three devices on a single wireless network.*

have a median latency ratio of greater than 0.2, meaning that for 30% of devices, at least half of the flows have end-to-end latencies where the home wireless network contributes more than 20% of the overall end-to-end latency.

These results generally suggest that *the RTT introduced by the wireless network may often be a significant fraction of the end-to-end RTT*. This finding is particularly meaningful in light of the many recent efforts by service providers to reduce latency to end-to-end services with myriad optimizations and careful placement of content. We recommend that, in addition to the attention that is already being paid to optimizing wide-area performance and host TCP connection settings, operators should also spend effort to improve home wireless network performance. A future avenue for research would be to understand the underlying causes of this latency, which may be due to channel contention, retransmission, or buffering delays caused by a bottlenecked wireless channel.

5.4 Devices Have Variable Performance

We also studied the extent to which wireless performance varies across devices in the same home network. We found many cases where the median wireless retransmission rates, ρ, for a device were higher than 0.1 (the threshold above which WTF deems that the client has a poor wireless channel). For the devices in the home shown in Figure 18, nearly all of the devices have $\rho > 0.1$ at least half of the time. Interestingly, Device 8 experiences a poor wireless channel nearly all of the time, suggesting a persistent problem that may result from device placement, interactions between the access point and that device's driver, or some other cause. Figure 19 shows the distribution of bitrates for three devices in a single home network. These devices consistently achieve much less than the maximum bitrate supported by the 802.11n protocol, and some devices perform considerably more poorly than others. In future work, we intend to explore these pathological cases of devices that consistently perform worse than others with more in-depth root cause analysis.

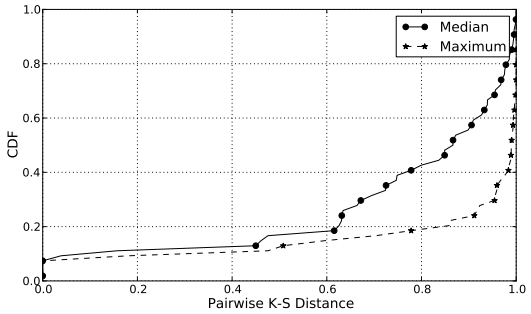


Figure 20: *Average K-S distance for distributions of raw bitrates between pairwise devices within a home network, for all home networks.*

Both Figures 18 and 19 show that different devices in the same home can experience different performance. To study this phenomenon, we measure the K-S distance of the distributions of raw wireless bitrates between each pair of devices in each home. Figure 20 plots the median and the maximum pairwise K-S distance in each home. We find that more than 80% of homes have at least one pair of devices with a K-S distance of more than 0.6, indicating that most homes have at least one poorly performing device (due to either poor placement, poor hardware, or poor drivers). Future work could involve investigating the disparate performance across devices further and determining whether the variability in device performance is caused by any single factor in particular.

6. CONCLUSION

We introduced WTF, a tool that runs on the router in a user's home network, that can provide visibility about whether performance problems exist inside the home network or elsewhere. Our results from 64 homes suggest when downstream throughput of a user's access link exceeds about 15 Mbits/s, the underlying cause of poor performance is more likely to be a poorly performing wireless network (when the downstream throughput exceeds 35 Mbits/s, the access network is never the problem). We also found that the 5 GHz spectrum range considerably outperforms the 2.4 GHz range, that TCP round-trip times inside home networks can often be very high, and that devices can exhibit extremely variable performance, even within a single home network. In addition to expanding our current deployment, we also plan to

develop techniques to better understand the causes of poor wireless performance in home networks.

This study opens several avenues for future work. First, although WTF can tell a user that their home wireless network is performing poorly, it does not offer any insights into the underlying causes. There is an acute need for methods that explain *why* various wireless performance problems exist in addition to where they are. Second, a follow-up to WTF could attribute problems that home network users experience to a more complete and more specific set of causes: for example, end hosts and applications can sometimes introduce performance problems. A more complete diagnosis tool might also identify whether problems truly lie in the access ISP or further afield in the wide area.

REFERENCES

[1] tcptrace: A TCP Connection Analysis Tool. `http://irg.cs.ohiou.edu/software/tcptrace/`.

[2] A. Adya, P. Bahl, R. Chandra, and L. Qiu. Architecture and techniques for diagnosing faults in ieee 802.11 infrastructure networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking*, MobiCom '04, pages 30–44, New York, NY, USA, 2004. ACM.

[3] B. Aggarwal, R. Bhagwan, T. Das, S. Eswaran, V. N. Padmanabhan, and G. M. Voelker. Netprints: diagnosing home network misconfigurations using shared knowledge. In *Proceedings of the 6th USENIX symposium on Networked systems design and implementation*, NSDI'09, pages 349–364, Berkeley, CA, USA, 2009. USENIX Association.

[4] N. Ahmed, U. Ismail, S. Keshav, and K. Papagiannaki. Online estimation of rf interference. In *Proceedings of the 2008 ACM CoNEXT Conference*, CoNEXT '08, pages 4:1–4:12, New York, NY, USA, 2008. ACM.

[5] D. Antoniades, M. Athanatos, A. Papadogiannakis, E. Markatos, and C. Dovrolis. Available bandwidth measurement as simple as running wget. In *Proc. of Passive and Active Measurement Conference (PAM 2006)*, pages 61–70. Citeseer, 2006.

[6] S. Biaz and N. H. Vaidya. Discriminating congestion losses from wireless losses using inter-arrival times at the receiver. In *Proceedings of the 1999 IEEE Symposium on Application - Specific Systems and Software Engineering and Technology*, ASSET '99, Washington, DC, USA, 1999. IEEE Computer Society.

[7] Y. Cheng, J. Bellardo, P. Benko, A. C. Snoeren, G. M. Voelker, and S. Savage. Jigsaw: Solving the puzzle of enterprise 802.11 analysis. In *Proc. ACM SIGCOMM*, Pisa, Italy, Aug. 2006.

[8] Y.-C. Cheng, M. Afanasyev, P. Verkaik, P. Benkö, J. Chiang, A. C. Snoeren, S. Savage, and G. M. Voelker. Automating cross-layer diagnosis of enterprise wireless networks. *SIGCOMM Comput. Commun. Rev.*, 37(4):25–36, Aug. 2007.

[9] M. Dischinger, M. Marcon, S. Guha, K. Gummadi, R. Mahajan, and S. Saroiu. Glasnost: Enabling end users to detect traffic differentiation. In *Proceedings of the 7th USENIX conference on Networked systems design and implementation*, pages 27–27. USENIX Association, 2010.

[10] Glasnost: Bringing Transparency to the Internet. `http://broadband.mpi-sws.mpg.de/transparency`.

[11] N. Hu, L. Li, Z. Mao, P. Steenkiste, and J. Wang. A measurement study of internet bottlenecks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 3, pages 1689 – 1700 vol. 3, march 2005.

[12] N. Hu, L. E. Li, and Z. M. Mao. Locating Internet bottlenecks: Algorithms, measurements, and implications. In *Proc. ACM SIGCOMM*, pages 41–54, Portland, OR, Aug. 2004.

[13] M. Jain and C. Dovrolis. Pathload: A measurement tool for end-to-end available bandwidth. In *In Proceedings of Passive and Active Measurements (PAM) Workshop*, pages 14–25, 2002.

[14] G. Judd and P. Steenkiste. Understanding Link-level 802.11 Behavior: Replacing Convention with Measurement. In *Wireless Internet Conference 2007 (Wicon07)*, Austin, TX, Oct. 2007.

[15] P. Kanuparthy and C. Dovrolis. Diffprobe: detecting isp service discrimination. In *Proceedings of the 29th conference on Information communications*, INFOCOM'10, pages 1649–1657, Piscataway, NJ, USA, 2010. IEEE Press.

[16] P. Kanuparthy, C. Dovrolis, and M. Ammar. Spectral probing, crosstalk and frequency multiplexing in internet paths. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, IMC '08, pages 291–304, New York, NY, USA, 2008. ACM.

[17] P. Kanuparthy, C. Dovrolis, K. Papagiannaki, S. Seshan, and P. Steenkiste. Can user-level probing detect and diagnose common home-wlan pathologies. *SIGCOMM Comput. Commun. Rev.*, 42(1):7–15, Jan. 2012.

[18] D. Katabi and C. Blake. Inferring congestion sharing and path characteristics from packet interarrival times. Technical Report MIT-LCS-TR-828, Massachusetts Institute of Technology, 2002.

[19] K. Lai and M. Baker. Nettimer: A tool for measuring bottleneck link bandwidth. In *Proceedings of the USENIX Symposium on Internet Technologies and Systems*, volume 134, 2001.

[20] K. Lakshminarayanan, S. Sapra, S. Seshan, and P. Steenkiste. Rfdump: an architecture for monitoring the wireless ether. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, CoNEXT '09, pages 253–264, 2009.

[21] R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan. Analyzing the mac-level behavior of wireless networks in the wild. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '06, pages 75–86, 2006.

[22] D. Niculescu. Interference map for 802.11 networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 339–350, New York, NY, USA, 2007. ACM.

[23] Radiotap. `http://radiotap.org`.

[24] S. Rayanchu, A. Mishra, D. Agrawal, S. Saha, and S. Banerjee. Diagnosing wireless packet losses in 802.11: Separating collision from weak signal. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 735 –743, april 2008.

[25] S. Rayanchu, A. Patro, and S. Banerjee. Airshark: detecting non-wifi rf devices using commodity wifi hardware. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, IMC '11, pages 137–154, New York, NY, USA, 2011. ACM.

[26] S. Rayanchu, A. Patro, and S. Banerjee. Catching whales and minnows using wifinet: deconstructing non-wifi interference using wifi hardware. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, pages 5–5, Berkeley, CA, USA, 2012. USENIX Association.

[27] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cottrell. pathchirp: Efficient available bandwidth estimation for network paths. In *Passive and active measurement workshop*, volume 4, 2003.

[28] S. Saroiu, P. Gummadi, and S. Gribble. Sprobe: A fast technique for measuring bottleneck bandwidth in uncooperative environments. In *IEEE INFOCOM*, page 1, 2002.

[29] S. Savage. Sting: a tcp-based network measurement tool. In *Proceedings of the 1999 USENIX Symposium on Internet Technologies and Systems*, pages 71–79, 1999.

[30] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè. Broadband Internet performance: A view from the gateway. In *Proceedings of the 2011 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '11, 2011.

[31] M. Tariq, M. Motiwala, N. Feamster, and M. Ammar. Detecting network neutrality violations with causal inference. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 289–300. ACM, 2009.

[32] Y. Zhang, L. Breslau, V. Paxson, and S. Shenker. On the characteristics and origins of internet flow rates. In *Proc. ACM SIGCOMM*, Pittsburgh, PA, Aug. 2002.