Pinpointing Home and Access Network Delays and Losses Using WiFi Neighbors

Lucas DiCioccio\(^\lambda\) Renata Teixeira\(^\ast\) Jim Kurose\(^\ddagger\) Martin May\(^\dagger\)
\(^\lambda\)Online.net \(^\ast\) INRIA \(^\ddagger\) University of Massachusetts, Amherst \(^\dagger\)Technicolor

ABSTRACT

Home Internet users and Internet access providers need tools to assist them in diagnosing and troubleshooting network performance problems. Today, expert users may rely on simple techniques using round-trip measurements to local and remote points to locate delays and losses on an end-to-end path. Unfortunately, round-trip measurements do not provide accurate diagnoses in the presence of asymmetric link capacities and performance, which is often the case in residential access. This paper introduces neighbor-assisted network diagnosis - an approach for pinpointing the location of delays and losses (among the home, access, and wide-area network), leveraging end-host multi-homing capabilities. We propose two instances of neighbor-assisted network diagnosis, NADD to diagnose delays and NALD for losses, that run on an end host connected simultaneously to the home gateway and to a neighbor WiFi access point. Our evaluation shows that NADD and NALD efficiently detect and distinguish uplink and downlink delays and losses with small error. In addition, we learn from a proof-of-concept deployment in five homes in France that our techniques can work “in the wild.”

1. INTRODUCTION

Hundreds of millions of homes today connect to the Internet using broadband access. Within these homes, numerous devices interconnect via a home network using technologies such as Ethernet, Powerline, MoCa, or WiFi. These home networks, in turn, connect to the larger Internet via access networks including DSL, cable, and wireless. Given this network heterogeneity and complexity, it is not surprising that end users can have difficulty in diagnosing network performance degradation and variability, and pinpointing the source of a problem at the network’s edge. Operators of access networks are also often under-equipped to diagnose performance degradation their customers experience, mainly due to the lack of a measurement vantage point inside users’ homes. While many tools exist for diagnosing network backbone performance problems (e.g., \([5, 22]\)), and for measuring end-system available bandwidth and capacity (e.g., \([12, 18, 20]\)), many fewer tools exist for diagnosing performance problems within the home network or within the access network. Typically, the only recourse is to employ generic tools such as ping, traceroute, and pathchar.

In this paper, we develop, analyze, prototype and evaluate new techniques for performing neighbor-assisted network diagnosis using probes sent from a multi-homed end-system to diagnose problems such as performance impairments and network failures. We present two such techniques, NADD (Neighbor-Assisted Delay Diagnosis) and NALD (Neighbor-Assisted Loss Diagnosis), that pinpoint delays and packet losses, respectively, in the home, neighbor network, and access networks. We use different types of probes from simple pings to the local home gateway interfaces and WiFi access point, to more complex cyclic probes sent from the end-host through one access network and then back to the multi-homed end-host via a second access network. Our evaluation in controlled experiments shows that NADD and NALD accurately diagnose where impairments occur (e.g., in the home or in the access network) as well as the direction (upstream or downstream) in which they occur. Additionally, our proof-of-concept deployment in five homes in Paris indicates that NADD and NALD work in practice and could be deployed at large scale. NADD and NALD are, more generally, applicable in any multi-homed scenario, making them new and important additions to the arsenal of network diagnostic techniques.

Since a multi-homed end-system is a prerequisite for the use of techniques such as NADD and NALD we begin this paper in Section 2 with a discussion of multi-homing opportunities. An end system may be multi-homed in many different ways, e.g., via a wired or WiFi access through one provider and 3G access through another provider. Alterna-
tively, as shown in Figure 1, an end system may access the Internet not only through its contracted ISP, but also through a neighboring community WiFi network. We thus characterize the availability of country-wide community WiFi networks deployed by French residential ISPs, finding that more than half the homes have access to at least one community WiFi network.

Section 3 describes the techniques employed in NADD to determine delays in network segments under investigation. We will see that multi-homing enables measurements such as RTT pings and cyclic measurement probes to be used to infer asymmetric network-internal delays and losses in the access network. We also discuss the technical challenges faced in implementing NADD as a tool.

Sections 4 and 5 detail our validation methodology and results, respectively, for NADD. Our controlled experiments, including a DSLAM access network and multiple Ethernet home networks, show that NADD detects and distinguishes delays introduced into each of the segments, and the upstream/downstream direction of delays when delays in the LAN and in the WiFi are symmetric. We also report on our proof-of-concept NADD deployment "in the wild" in five homes in France. Section 6 describes and evaluates NALD, a technique parallel to NADD that characterizes losses rather than delays.

End-users, ISPs, and over-the-top service providers may all take advantage of our techniques. We discuss such possible NADD and NALD deployments in Section 7. Section 8 presents related work.

2. MOTIVATION: MULTI-HOMING IN HOME NETWORKS

Several past efforts have proposed to take advantage of multi-homing in order to increase network resilience [1] or aggregate network capacity [16, 27, 30]. In this paper, we take advantage of multi-homing to diagnose network delays and losses in the access network and in the home network. In this section, we discuss the increasing opportunities for multi-homing in today’s networks, and then assess the potential of multi-homing via neighbor WiFi networks using WiFi scan results collected in France.

2.1 Opportunities for multi-homing

Today’s Internet users can access the Internet from various devices using multiple access technologies, including DSL, cable, and 3G. The co-existence of these technologies in home networks presents opportunities for multi-homing. An increasingly common configuration of multi-homing is a home computer connected via a cable or DSL network to the Internet, and a wireless tablet or phone connected to the Internet via 3G. In this scenario, a wireless phone can connect to the Internet using either the 3G connection or the home network. Other devices can also benefit from a 3G connection via tethering, allowing, for example, a computer connected via Ethernet to the DSL gateway to also connect over 3G via the smartphone. Multi-homing is also possible via strictly wired connections - a user may subscribe to both cable and an ADSL connection. More typically, however, a household will subscribe to only a single Internet access provider.

Even when users subscribe to a single access provider, multi-homing may be accomplished via a neighbor’s wireless access network. Some users open their WiFi network (as shown in Han’s “wardriving” study [17]). Others may participate in a WiFi community, where community users have credentials allowing access to one another’s WiFi access points. FON [13] was the first user-initiated WiFi community with large-scale adoption. ISPs also provide WiFi community as a service. For example, most access ISPs in France deploy a WiFi community, providing an ISP’s customers with credentials to connect to WiFi access points of other customers of that ISP. Hence, customers can benefit from “free” WiFi access in most of France. ISPs in other countries also deploy WiFi communities: for instance, BT Wi-Fi from British Telecom in the UK or XFINITY WiFi from Comcast in the United States. Indeed, we are seeing more and more ISPs providing community WiFi as the capacity of home access links increase.

In this paper, we focus on the use of WiFi as one of the multi-homing access technologies. The techniques behind NADD and NALD, however, are applicable in any multi-homed end-host setting. As a motivation that neighbor-assisted diagnosis can potentially be deployed today at large scale, we now characterize the prevalence of different WiFi communities in France.

2.2 Prevalence of WiFi communities

We evaluate the prevalence of WiFi communities in France with data collected using HomeNet Profiler [11], a home network measurement tool that volunteers run from their home computers. Since April 4th, 2011 users ran the WiFi tests of HomeNet Profiler in 692 homes in France, mostly in large cities such as Paris, Lyon, and Marseille. We refer to access points advertising one of the WiFi communities as community access points. We study how often we observe such community access points in the results of HomeNet Profiler’s WiFi scans. In particular we study the most common WiFi communities in France: ‘FreeWifi’, ‘freephonie’, ‘Bouygues Telecom WiFi’, ‘Neuf WiFi’, and ‘SFR WiFi Public’. Freephonie and FreeWi are two communities of the same ISP called Free. Freephonie only shares the access link for phones whereas FreeWifi provides Internet service. Similarly, SFR WiFi and Neuf WiFi belong to SFR. In France, Orange WiFi reports 30,000 hotspots1 and SFR WiFi reports four millions hotspots.2 Other French WiFi communities do not disclose the number of participants.

Annotations:
1 http://www.orange-wifi.com/
2 http://www.sfr.fr/telephonie-mobile/services-options/services-smartphones/reseau-wifi/
connected to the Internet via two interfaces with distinct access networks (one typically being its own home network, and one being a neighboring wireless network) to determine where, and in which direction (upstream or downstream) delays occur in its connectivity to the edge of the Internet.

We first describe our network setting, as well as a simple (but as we will see, often inaccurate) ping-based approach for pinpointing the location of delay. We then describe NADD, discuss the technical challenges when implementing this approach, and detail our implementation.

3.1 Network setting

Figure 3 illustrates our network setting. An end system has a primary connection to the Internet through its “home” network. An interface on the end system (labeled a in Figure 3) connects via its home network to an interface on its home gateway router (b) which then connects via interface c to the access network and from there to the larger Internet. A user experiencing congestion (e.g., as manifested in high delay) is interested in where this delay occurs - is the delay in the home network or in the access network, and does this delay occur in the upstream or downstream direction? The user’s ISP is also interested in this information, particularly when the user calls customer support to report poor performance.

In order to perform this diagnosis, NADD leverages the circumstances in which the user has a neighbor network that also provides Internet connectivity. In this case, the end system connects via interface f to interface e in the neighbor network, and from there to the larger Internet. As discussed in the previous section, this second avenue of connectivity can often be found (particularly in urban settings) when an end system has access to a WiFi community network. The home network itself may be wired or wireless (although in the wireless case, the end system would require two separate wireless network interfaces (an atypical scenario in practice). For simplicity and maximal applicability, we consider here the case where the a-to-b network is wired Ethernet and the f-to-e network is wireless. We emphasize, however, the general technique behind NADD is independent of network type.

3.2 Estimating delays via landmark pings

Perhaps the simplest way to estimate delays is to use a tool such as traceroute or ping or their variants to measure the round-trip delay from the end host to a landmark and assume that delays in the upstream and downstream directions are symmetric. Using this technique, for example, the end host can ping the path aba and measure the round trip delay \( d_{aba} \). Here we use the hat notation to indicate a measured value and use the subscript to indicate the path measured. Assuming symmetric upstream and downstream delays, i.e., that the a-to-b delay, denoted \( d_{ab} \), equals the b-to-a delay, \( d_{ba} \), these delays are then simply estimated as \( d_{ab} = d_{ba} = \hat{d}_{aba}/2 \). Given the estimates \( \hat{d}_{aba} \), the end host
3.3 Neighbor-assisted delay diagnosis

NADD exploits the fact that hosts with two network interfaces have a cyclic topology with three segments (six directional segments, \(ab, cd, ef, fe, dc, ba\)), as shown in Figure 3, and can send probe/measurement packets from one interface to another through these segments.

Given the ability to send probe/measurement packets along a directional path (e.g., along path \(abcdef\)), it might be tempting to use IP timestamp options to directly measure directional delays at each hop along the path. Prior studies [9, 24], however, document the relatively sparse implementation of IP timestamp options in wired backbone routers, particularly when the option request is to record a timestamp at a given IP interface en route to a final destination. Indeed, our own measurements (see Table 3 in Section 5) indicate that fewer than 10% of the packets sent along the cyclic path \(abcdef\) or \(fedcba\) in five different home network settings actually completed the cycle with the requested timestamp. Thus, the use of timestamps for delay diagnosis in edge networks appears even more problematic than (the already rather bleak situation) in wired backbone networks.

Given that individual, per-hop delays are not directly measurable with IP timestamp options, is all hope lost of determining these per-hop delays and thus pinpointing and diagnosing delays? Fortunately, the answer is no! NADD provides a methodology for measuring round-trip and cyclic delays in a setting such as Figure 3, and then inferring these per-hop delays.

Figure 3: Notations for NADD

![Diagram of network topology]

The host can individually ping interfaces \(b\) and \(e\), obtaining measurements \(d_{aba}\) and \(d_{efe}\).

- **RTT estimates to the public (i.e., Internet) interface of the local gateway via ping.** For example, to ping the public interface of the home network gateway, the host can send a ping via the community WiFi interface along the path \(f_{edc}\), yielding the RTT measurement \(d_{fedcdef}\). Note that to do this, the end system must first discover the IP address of interface \(c\).

- **Clockwise and counter-clockwise full cycle probes.** With some effort (described below), the host can send itself a probe message that traverses a cycle (either clockwise or counter-clockwise) beginning and ending at the host itself. For example, by sending a probe on outgoing interface \(a\) addressed to its own interface \(f\), the host can directly measure the counter-clockwise cyclic one way delay \(d_{abcdef}\). The clockwise cyclic delay \(d_{fedcba}\) can similarly be measured.

As we discuss in Section 5.2, these measurements can vary even in short periods of time. We estimate these values by sending a train of probes and taking the minimum value. As the minimum delay represents the best case a user would experience, taking the minimum delay of a train of probes is a conservative choice to filter-out transient delays. We call one round of measurements the operation of estimating all six delays with trains of probes. Note that in each of these cases, the measured multihop delays are composed of various combinations of the (unknown) per-hop delays \(d_{ab}, d_{ba}, d_{cd}, d_{dc}, d_{fe}, d_{ef}\). Given the topology in Figure 3, we can express these relationships using the following system of linear equations:

\[
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 \\
\end{pmatrix}
\begin{pmatrix}
d_{ab} \\
d_{ba} \\
d_{cd} \\
d_{dc} \\
d_{fe} \\
d_{ef} \\
\end{pmatrix}
= 
\begin{pmatrix}
d_{faba} \\
d_{abcdef} \\
d_{edcdef} \\
d_{fedcba} \\
d_{fedcba} \\
\end{pmatrix}
\]

This system of six equations has six unknowns. Unfortunately, the rank of the matrix is four. In order to solve this system, we will thus need either to reduce the number of unknowns or to increase the number of equations. One way to do this is to assume symmetric delays in the home network and in the WiFi neighbor (i.e., \(d_{ab} = d_{ba}\) and \(d_{fe} = d_{ef}\)). This assumption is widely accepted for Ethernet links, but does not always hold for WiFi networks. For example, a study of VoIP over wireless LANs [28] shows that an increased number of VoIP flows increase the downlink delay whereas the uplink delay remains small. Yet, we consider this assumption reasonable in our case, because network delays in Ethernet or WiFi are typically smaller than access network delays given link transmissions speeds (as we confirm in our deployment, see results in Figure 6). We specifically do not assume that access links are symmetric, as this...
is generally not the case with home Internet access. Assuming symmetric network delays in the home and to the neighbor WiFi reduces the number of unknowns to four, yielding six measurement equations with now four unknowns.

With more equations than unknowns, we now have a choice of which set of measurement equations to use. One reduction uses both one-way measurements (i.e., \(d_{abcdef}\) and \(d_{fedcba}\)):

\[
\begin{pmatrix}
2 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 \\
1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1
\end{pmatrix} \cdot \begin{pmatrix}
d_{aba} \\
d_{acd} \\
d_{dec} \\
d_{fde}
\end{pmatrix} = \begin{pmatrix}
d_{aba} \\
d_{fca} \\
d_{abcdef} \\
d_{fedcba}
\end{pmatrix}
\]

A second reduction uses an RTT measurement to a public gateway interface instead of a one-way cyclic delay measurement. Specifically, replacing the last equation above yields:

\[
\begin{pmatrix}
2 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 \\
1 & 1 & 0 & 1 \\
0 & 1 & 1 & 2
\end{pmatrix} \cdot \begin{pmatrix}
d_{aba} \\
d_{acd} \\
d_{dec} \\
d_{fde}
\end{pmatrix} = \begin{pmatrix}
d_{aba} \\
d_{fca} \\
d_{abcdef} \\
d_{fedcba}
\end{pmatrix}
\]

Depending on which equations are actually used to solve for these four unknowns, NADD will give slightly different performance estimates, a topic we’ll investigate in Section 5.

3.4 Measurement implementation

Network address translators (NATs) complicate our measurement technique. End-hosts generally receive a private IP address (for example, in the 192.168/16 subnet) and connect to the Internet through a NAT. An end-host must first discover the public IP addresses of both gateways to probe them. We gather the public IP addresses of home gateways with the support of a remote web server that returns the requesting IP address of a web client.

NAT also introduces complications into the one-way cyclic delay measurements. Here, the problem is that a NATed host must be able to receive packets on one of its interfaces. In order for the incoming measurement packets to punch through the NAT, the NAT must support port mapping for incoming packets. To accomplish this, NADD automatically configures a port-mapping between a port on the home gateway and a port on the measurement laptop using the UPnP protocol [31]; NADD accomplishes this without user intervention. Our deployment maps port 53 (DNS) because both FreeWiFi and SFRWiFi allow UDP packets to be sent to this port. We have found that the community access points do not support UPnP port-mapping. Consequently, we can only initiate cycle measurements from the WiFi interface in a clockwise direction in Figure 3.

Our measurement implementation requires little support from the home gateways. Both gateways must reply to probes (e.g., ICMP pings or ICMP TTL-expired traceroute) used to measure a round-trip delay and one of the gateways must support port-mapping configuration via UPnP. UPnP can also alleviate the need for a remote server to discover public IP addresses. Our earlier study observed UPnP-enabled gateways in 35% of homes [10]. Our analysis of the HomeNet Profiler dataset shows that in more than 90% of the 2381 homes with an UPnP-enabled gateway, the gateway also responds to ping-style probes. These results indicate that NADD requirements are reasonable in practice today. In Section 7 we discuss NADD’s possible deployment scenarios.

We next evaluate NADD with controlled experiments and with a proof-of-concept deployment.

4. EVALUATION METHODOLOGY

This section describes the methodology used to evaluate NADD. We perform both controlled experiments in a testbed that emulates home networks connected to one residential access ISP as well as a proof-of-concept deployment and measurement. The controlled experiments allow us to evaluate the accuracy of our delay inferences in various scenarios, knowing the ground truth values. The deployment allows us to experiment with real delays and delay variations, even though we cannot confirm the accuracy of our inferences.

4.1 Controlled experiments

Testbed.

As illustrated in Figure 4, we created a fully-controlled testbed that emulates two home networks connected to an access ISP (one home network represents the home where our measurement host is connected to and the other represents the neighbor). We emulate the gateways in each home with two Linux computers each with an ADSL interface to emulate the access link and an Ethernet interface to emulate the home network. The measurement host, which is another Linux computer, connects to both gateways over Ethernet. Even though in the real-world deployment the link between the host and the neighbor gateway will be a wireless connection, we connect the measurement host to the neighbor gateway using Ethernet to have better control of the introduced delays. We add delays in the home network using netem on each gateway. The ADSL interface is connected to an ADSL2+ DSLAM. Each home gateway sets up an Internet connection with PPP over an Ethernet tunnel over ATM to a FreeBSD server. This server runs dummynet, allowing us to introduce access link delays.

Parameter settings.
We determine the measurement capabilities of FreeWifi and SFRWifi (i.e., which kinds of measurements can a guest host perform when connected to these community access points) from two homes in Paris. From each home we connect to different FreeWifi and SFRWifi access points. We use Netalzyr [18] to perform a complete test of the capabilities of each community access point. We repeat each test twice and found that our results are consistent over time. Policies for a guest host to access these community access points have been stable from November 2011 to May 2014.

Table 1 summarizes the measurement capabilities of FreeWifi and SFRWifi. On FreeWifi, the guest host obtains a public IP address half of the time. Otherwise, the guest host obtains a private IP address. We didn’t identify patterns that might explain whether the guest host receives a public or a private IP address. When receiving a public IP address, the guest is able to receive incoming connections, can perform `traceroute`, and experiences no port-based protocol filtering. When the guest host obtains a private IP address on FreeWifi, however, it cannot send UDP probes to arbitrary IP Internet addresses. On FreeWifi, the guest host bandwidth is capped at 1 Mbps for download; the upload bandwidth is not capped (it is hence limited by the WiFi connectivity and the access link). Although we didn’t identify any pattern of bandwidth shaping over SFRWifi, a guest host connected to SFRWifi has many more limitations than when connecting with FreeWifi. SFRWifi filters many protocols and `traceroute` is also blocked past the SFRWifi access point (i.e., the guest host only receives the response for the first hop; no other responses are forwarded to the guest host).

FreeWifi permissible policies when hosts obtain a public IP address are ideal for neighbor-assisted diagnosis; but our implementation works on both SFRWifi and FreeWifi provided that the user’s home gateway enables UPnP.

### 4.2.2 Real-World Measurements

We deploy notebooks as measurement hosts in five homes in Paris, France. We recruit volunteers in Paris because WiFi communities are particularly prevalent in Paris. Measurement hosts are directly connected to the home gateway. We use a server at UPMC, which is also in Paris, to control the measurement hosts and collect measurement reports. Our goal with these real-world measurements is not to provide a full-fledged characterization of delays in access links. We instead want to evaluate NADD and understand the practical limitations of NADD using existing WiFi communities. Each measurement host tries to connect to neighbor access points advertising both FreeWifi and SFRWifi every 30 minutes, not to overload the volunteer’s network. We develop scripts to automatically log into the FreeWifi and SFRWifi authentication portals. The measurement hosts are configured with the credentials to connect to both of these communities.

For each successful connection to a WiFi neighbor, we perform a number of measurement rounds with different pa-
parameters. For each round, the end-host sends six trains of probes to estimate each of the six delays used in NADD in the following order: $d_{ef}$, $d_{fedeef}$, $d_{ab}$, $d_{abedeba}$, $d_{fede}$. First, we send a train of probes back-to-back to estimate $d_{ef}$, we wait up to five seconds to receive all the responses, then we send the next probe train to measure $d_{fedeef}$, wait up to five seconds, and so on. We configure probe trains with different types of probes, probe sizes, and number of probes per train. We set the number of probes in a train to either 3 or 20; and the probe size to 64 bytes (i.e., a small packet) and 1400 bytes (a large packet). We measure full cycle probes with UDP and RTT estimates with ICMP. For the full cycle probes, we measure with and without requesting the IP timestamp of the home gateway and the neighbor gateway. A measurement host first performs a round with eleven trains of small probes per delay measurement (ten trains of 3 probes and one of 20 probes), then eleven trains of large probes (again ten trains with 3 probes and one with 20 probes). We then run the same 22 trains with the IP timestamp request.

We run measurement rounds from the five homes for nine days in Jul. 2012 and in Dec. 2012. We collect a total of 1268 measurement rounds in this period, but only 339 (27%) of these measurement rounds were successful. The authentication portals of FreeWiFi and SFRWiFi often refuse access to the laptops of our proof-of-concept deployment (for instance, if portals consider that credentials are already in use). Note that this authentication problem happens in our deployment because we were using credentials that people put publicly-available for anyone to use. This problem would not happen if we had used users in each home had used their personal credentials. Each measurement host successfully connects to between two and six different neighbor WiFi networks.

5. EVALUATION RESULTS

This section first evaluates the accuracy of NADD with controlled experiments and then discusses how to address issues such as lost probes and measurement consistency that arise in real deployments.

5.1 Validation

We use the testbed described in Section 4.1 to evaluate NADD’s accuracy in a fully controlled setting. We evaluate NADD using the five different possible reductions to the system of linear equations in Equation 1. Our results show that NADD infers the same delays for all reductions in all scenarios within a range of 5 ms. Given that the different versions of the equation system lead to similar results, we only present results using the system presented in Equation 2, which uses the two one-way cycle measurements.

Table 2 presents the median absolute error out of 100 measurements for NADD and landmark pings in four representative scenarios in our controlled testbed. Results for other scenarios are similar. In this testbed, we add delays into network segments $ab$, $ba$, $cd$, $dc$ (refer back to Figure 3 for a description of link notation). The first four rows in Table 2 correspond to cases where delays in the upstream and downstream direction of each link are symmetric, whereas the last four rows correspond to asymmetric delays in the LAN or the access link. We present the absolute error of NADD and landmark pings to estimate delays in the LAN ($d_{ab}$ and $d_{ba}$) and in the access link ($d_{cd}$ and $d_{dc}$). When delays are symmetric, both landmark pings and NADD perform well. The absolute error is less than 1 ms in all cases in the first four rows of Table 2.

Not surprisingly, when we violate NADD’s assumption of symmetric delays in the home network (row labeled $ab = 30$ ms), NADD performs poorly. The absolute error is approximately +/-15 ms for the access and the LAN. This error corresponds to half of the delay difference between the upstream and downstream directions of the LAN, so in networks with higher asymmetry we expect larger absolute errors. Landmark pings have the same +/-15 ms error in the LAN, but practically no error in the access.

We do not expect highly asymmetric delays to occur in practice in home networks, since these networks are implemented with Ethernet, WiFi, or Powerline, which have equal upstream and downstream capacity and high transmission rates. Even if WiFi contention or buffering at the end-host or gateway creates some asymmetry, in practice this asymmetry will be small. As we’ll see in the next section, typical home network delays (or $d_{aba}$) vary between 1 ms and 5 ms, while access network delays (or $d_{abedeba} - d_{aba}$) are an order of magnitude higher. Hence, even if there is asymmetry in the home network, the absolute error should be small. We are currently investigating a method to detect that NADD’s assumptions are violated. The idea is to perform NADD rounds periodically and search for large (negative) variations in the time series of the inferred delay of each network segment. The inference of such a negative delay would indicate that our assumptions are likely violated.

NADD shines when delay asymmetry occurs in the access network (for example, when $cd = 100$ ms in Table 2). NADD infers correct delays with less than one millisecond of error. Meanwhile, the landmark ping technique incorrectly attributes half of the delay to each direction. We expect this scenario to be the most common in practice, because most residential ISPs offer higher downstream than

## Table 2: Error of NADD and landmark ping

<table>
<thead>
<tr>
<th>Added Delay</th>
<th>Technique</th>
<th>Error (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Access</td>
<td>LAN</td>
</tr>
<tr>
<td>$ab = ba = 30$ ms</td>
<td>NADD</td>
<td>-0.1 -0.2</td>
</tr>
<tr>
<td></td>
<td>Landmark</td>
<td>-0.3 -0.3</td>
</tr>
<tr>
<td>$cd = dc = 100$ ms</td>
<td>NADD</td>
<td>-0.9 -1.0</td>
</tr>
<tr>
<td></td>
<td>Landmark</td>
<td>-0.7 -0.7</td>
</tr>
<tr>
<td>$ab = 30$ ms</td>
<td>NADD</td>
<td>15.4 -16.3</td>
</tr>
<tr>
<td></td>
<td>Landmark</td>
<td>-0.3 -0.3</td>
</tr>
<tr>
<td>$cd = 100$ ms</td>
<td>NADD</td>
<td>-0.4 -0.3</td>
</tr>
<tr>
<td></td>
<td>Landmark</td>
<td>-50.4 49.6</td>
</tr>
</tbody>
</table>
Probed segments | Probe success |
---|---|
ICMP | \( \hat{a}_b \) | 100 % |
ICMP | \( \hat{a}_b \) | 67 % |
UDP | \( \hat{a}_b \) | 67 % |
ICMP | \( \hat{a}_b \) | 66 % |
UDP (timestamp) | \( \hat{a}_b \) | 9 % |
UDP (timestamp) | \( \hat{a}_b \) | 9 % |

Table 3: Observed probe successes

upstream capacity and because buffering at the gateway (which can sometimes introduce seconds of delay [14, 18]) will affect access network delays rather than LAN delays.

The controlled experiments evaluate NADD’s accuracy in an ideal case. In practice, however, NADD will experience measurement noise — instances when measurement probes are lost or when delays vary considerably between two consecutive probes. We study these issues in the next section.

5.2 Effects of measurement noise

To evaluate the measurement noise that NADD must handle in practice, we analyze the measurements obtained from our proof-of-concept deployment described in Section 4.2. The goal of the analysis in this section is to demonstrate the feasibility of NADD, a large-scale characterization is out of the scope of this paper. Our analysis of the different parameters of probe trains show that small probes (of 64 bytes) in trains of three probes represent a good tradeoff between probing overhead and probe success. Hence, the rest of this section presents results for trains of three probes of 64 bytes. Our conclusions are similar with other parameter settings.

**Probes success**

Probes or their responses may be rate-limited, blocked, or lost in the network. Without valid responses to probes, NADD cannot infer delays. Table 3 presents the observed probability of success of the six types of probes that NADD uses in 339 distinct measurement rounds. The names of probed segments are shown in Figure 3. We define the probe success as the fraction of transmitted probes for which the measurement host receives a response. We compute the probe success over thousands of probe trials for each measurement. ICMP probes measure round trip delays, whereas UDP probes measure one-way delays. Although we don’t use probes with timestamps in NADD, we also present the probe success of these probes for reference.

The probe success depends on the probe type and the measured segment. Probes with timestamps are often dropped or blocked. Only 9% of the probes transmitted with timestamps yielded a response. As discussed in Section 3, this result prevented us from incorporating IP-options-timestamped probes in NADD. All ping to home gateways (\( \hat{d}_{\text{aba}} \)) succeed because measurement hosts are directly connected to their home gateway in Ethernet. Other measurements have probe success between 66% and 87% with the exception of the ping to the home gateway from the neighbor WiFi (\( \hat{d}_{\text{fedcba}} \)).

```
<table>
<thead>
<tr>
<th>Probe type</th>
<th>Probed segments</th>
<th>Probe success</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICMP</td>
<td>( \hat{a}_b )</td>
<td>100 %</td>
</tr>
<tr>
<td>ICMP</td>
<td>( \hat{a}_b )</td>
<td>67 %</td>
</tr>
<tr>
<td>UDP</td>
<td>( \hat{a}_b )</td>
<td>67 %</td>
</tr>
<tr>
<td>ICMP</td>
<td>( \hat{a}_b )</td>
<td>66 %</td>
</tr>
<tr>
<td>UDP (timestamp)</td>
<td>( \hat{a}_b )</td>
<td>9 %</td>
</tr>
<tr>
<td>UDP (timestamp)</td>
<td>( \hat{a}_b )</td>
<td>9 %</td>
</tr>
</tbody>
</table>
```

Figure 5: Delay variability in the deployment

Our inspection of the traces shows that higher loss rates happen as a result of two independent issues: SFRWiFi blocks ICMP probes and one home gateway dropped all ICMP ping requests to its public interface. After removing these measurements, the probe success of \( \hat{d}_{\text{fedcba}} \) is 62%.

These results suggest that NADD can already be applied in practice. Even though some of the probes are blocked in our proof-of-concept, NADD is still able to solve the linear equations in many cases, because it only requires four out of the six probes to be successful. Out of 339 measurement rounds, the measurement host has enough measurements to solve the linear equations in 178 measurement rounds. It has all the six measurements in 84 rounds. Moreover, these results only reflect the current configuration of these community access points. If NADD were to be widely deployed, ISPs would have incentives to configure their network policies to allow these types of probes. We discuss further deployment considerations in Section 7.

**Consistency of measured delays**

NADD assumes that different probes traversing a given network segment will experience the same delay. In practice, consecutive probes measuring the same set of network segments may lead to different delay estimates, because of transient congestion. We study the consistency of measured delays by computing the delay variability over the ten trains of three back-to-back probes to a given set of network segments. We define the delay variability of a round as the delay difference between the probe that inferred the maximum delay and the one that inferred the minimum delay.

Figure 5 presents the cumulative density function of delay variability for three different sets of probed segments. The left-most curve corresponds to the LAN segment (\( \hat{d}_{\text{aba}} \)). In approximately 80% of trains the delay variability in the LAN is less than 1 ms, and the maximum variability in the LAN is 6.5 ms. Under such consistent delays NADD’s inference should be the most accurate. The two other curves, however, show that the delay variability in the WiFi ping (\( \hat{d}_{\text{fedc}} \)) and the cycle (\( \hat{d}_{\text{fedcba}} \)) are rarely less than 10 ms. The delay...
variability in these two segments can be as high as 900 ms. The distributions of delay variability of the WiFi ping and of the cycle are strikingly close. This observation leads us to believe that most of the delay variability in our proof-of-concept happens in the WiFi segment.

The high deviation between different delay measurements of one given segment implies that we can not assume that different probes that traverse the same segment will experience consistent delays. To address this issue, NADD uses the minimum delay out of the ten trains of three probes to populate the equation system. The minimum delay is usually more stable, because it avoids variability introduced by short-time-scale, transient congestion episodes.

Next, we study how the violation of the asymmetry assumption in the LAN or in the WiFi impacts the accuracy of NADD. Figure 6 shows the cumulative density function of raw estimates of $d_{aba}$ (LAN), $d_{abcdef}$ (WiFi), $d_{abcdef}$ (Access), and $d_{abcdef}$ (Round trip) using the minimum delay of ten trains of three probes. LAN delays are always below one millisecond and hence can be neglected. The WiFi delay is larger. We notice two strong modes in WiFi delay around 3 ms and 22 ms. These values most likely correspond to different WiFi rates. The measurement rounds in each of these two modes belong to distinct WiFi access points, with different signal strength. Finally, the cycle measurements crossing the access link range from 29 ms to 53 ms and the round-trip measurements crossing the access link are mostly above 50 ms. These values are in the same order of magnitude as WiFi delays with low signal strength. Hence, we conclude that the asymmetry in the WiFi or in the LAN should not introduce significant error, except when the end host connects to a WiFi neighbor with a low signal strength. Our analysis shows that these results do not change significantly when aggregating delays with the median delay. Using the maximum delay, however, the delay variability shown on Figure 5 strongly skews raw delay estimations. For instance, 38% of the trains of probes along $d_{abcdef}$ have a maximum delay larger than 100 ms.

**Consistency of inferred delay**

In 84 measurement rounds where we receive responses to all six measurements required in Equation 1, we infer delays of each network segment with the five reduced equation systems. We then compare whether the inferred delays of each network segment are consistent across the different reductions. We measure consistency with the delay variability computed as the delay difference between the maximum and the minimum inferred delay for a network segment out of the five inferred delays.

NADD’s inferred delays are consistent in our deployment. All solutions infer the same value of $d_{ab}$, since the single measurement $d_{aba}$ is itself sufficient to compute $d_{ab}$. The same argument applies to $d_{cf}$. The delay variability of $d_{cd}$ and $d_{dc}$ are strongly correlated (the Pearson coefficient of correlation is 0.94). The median delay variability of $d_{cd}$ and $d_{dc}$ is 1.7 ms and the 95th-percentile is 6.7 ms. In our dataset, we found only two distinct neighbor access points out of 10 where NADD inferred delays in the access segment with delay variability higher than 10 ms.

6. NALD: NEIGHBOR-ASSISTED LOSS DIAGNOSIS

In this section, we present a variation of NADD that infers losses, rather than delay, associated with a link (directionally). We call this variation “NALD” for Neighbor-Assisted Loss Diagnosis. We first describe NALD. Then, we describe our evaluation method and our evaluation with controlled experiments and in a real deployment.

6.1 Neighbor-assisted loss diagnosis

NALD also uses the network setting presented in Figure 3 to identify network segments with high packet losses. NALD uses the same types of probes used to measuring delays to infer losses. Estimating loss, however, is more challenging than estimating delay. First, estimating small packet losses with reasonable precision requires sending probe trains with a large number of probes; this will increase the measurement time. Second, sending probes back-to-back can induce congestion, which will bias the estimation of packet losses; to avoid this issue, we send probes spaced according to a Gamma process [3].

First, let us adapt the equation system from NADD in Equation 1 for loss. For simplicity, let us assume that packet losses on segments of a given path are independent. We’ll validate this assumption shortly in the deployment scenario from Section 4.2. A path’s transmission probability (i.e., the complement of the probability that a packet is lost on a path) will then be the product of the transmission probabilities of that path’s segments. Let $P_{ab}$ be the transmission probability of segment $ab$, and $P_{aba}$ the measured transmission probability from $a$ to $b$ and back to $a$. The logarithm of path transmission probability then becomes the sum of the loga-

![Figure 6: Raw delay estimations from measurements. One point represents the minimum delay out of ten trains of three probes.](image)
rithms of the individual segment transmission probabilities; adding up in a manner similar to delays. For example, we have \( \log(P_{aba}) = \log(P_{ab}) + \log(P_{ba}) \) for strictly positive transmission probabilities. Hence, we can use a system of equations similar to that of NADD to infer the probability of packet loss (expressed as a transmission probability) on each network segment as follows.

\[
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 1
\end{pmatrix} 
\begin{pmatrix}
\log(P_{ab}) \\
\log(P_{ba}) \\
\log(P_{cd}) \\
\log(P_{dc}) \\
\log(P_{ef}) \\
\log(P_{fe})
\end{pmatrix} = \begin{pmatrix}
\log(P_{aba}) \\
\log(P_{badeba}) \\
\log(P_{be}) \\
\log(P_{bede}) \\
\log(P_{be}) \\
\log(P_{fe})
\end{pmatrix}
\]

(4)

As with NADD, this system has six unknowns, but the rank of the matrix is four. To solve this system of equations, we make the assumption that packet loss probabilities are symmetric in the home network and on the WiFi link to the neighbor home gateway. This assumption is reasonable for wired technologies such as Ethernet in the home network, but it may not hold for some WiFi configurations. For example, hidden terminals might cause more losses from the client to the access point than on the reverse path. Under the loss symmetry assumption, we can reduce Equation 4 to five different equation systems equivalent to the ones we identify for NADD.

6.2 Evaluation method

Controlled experiments We evaluate NALD with the testbed described in Section 4.1 and consider cases with high losses. For the WiFi link to the neighbor gateway, we set the loss probability to 20%; and in the access network segment we set loss probability to 10%. We test both symmetric and asymmetric losses. We execute one repetition of NALD for each scenario. We assume that there are no losses other than what we introduce with netem and dummynet. We estimate losses with 100 probes of 64 bytes. The interval between two probes follow a Gamma process configured such that we send two probes per second on average. Thus, inferring the losses of a single path takes around two and a half minutes.

Deployment We use the deployment described in Section 4.2 to evaluate NALD in the wild. We evaluate two different probing schemes. From Oct. 2012 to Jan. 2013, we configured the deployment to send 300 probes per path and to probe paths sequentially (i.e., to send all probes to a path before probing the next path). This approach estimates loss probabilities with higher precision, but it takes a long time to measure each path, so network conditions may change between the time we start measuring the first path and we end measuring the last one. From Feb. 2013 to Jun. 2013, we configured the deployment to send ten probes per path interleaved, i.e., one probe to each of the six paths ten times. This scheme estimates loss probabilities with lower precision, but increases the likelihood that probes experience similar conditions. Despite the differences because of the lower precision of the latter scheme, we find that the distribution

<table>
<thead>
<tr>
<th>Target Losses</th>
<th>Technique</th>
<th>Error (percentile points)</th>
</tr>
</thead>
<tbody>
<tr>
<td>fe = ef = 20 %</td>
<td>Landmark</td>
<td>-1.4</td>
</tr>
<tr>
<td>cd = dc = 10 %</td>
<td>Landmark</td>
<td>0.1</td>
</tr>
<tr>
<td>fe = 20 %</td>
<td>Landmark</td>
<td>-11.3</td>
</tr>
<tr>
<td>cd = 10 %</td>
<td>Landmark</td>
<td>-5.1</td>
</tr>
</tbody>
</table>

Table 4: Absolute error (in percentile points) of NALD and landmark ping under symmetric and asymmetric losses

of loss probabilities for each of the network segments was similar when using these two probing schemes. Hence, we present results that combine data collected with both probing schemes in the next section.

6.3 Evaluation results

We first use the controlled experiments to evaluate NALD’s accuracy. Then, we use the deployment to test NALD’s assumption that the loss probabilities of segments of a path are independent and to study where losses occur for the five homes in the deployment.

6.3.1 Validation

We evaluate NALD’s accuracy using the controlled experiments described in Section 6.2. We compute absolute errors as the loss probability inferred by NALD minus the loss probability we introduce explicitly. For reference, we compare NALD with a solution that infers loss probabilities using landmark pings. We adapt the landmark ping solution described in Section 3.2 to infer loss probabilities by issuing 100 ping packets per probed path. Table 4 presents the median errors for 10 NALD rounds and for 10 rounds of landmark pings (using 100 packets). We evaluate four scenarios: losses on the access segment (cd or dc) and on the neighbor WiFi (ef or fe); and symmetric and asymmetric losses. We present the absolute error in percentile points. For example, when we set the loss probability on fe = ef to 20%, landmark pings infer a loss probability of 20.8%. Both landmark pings and NALD infer loss probability within less than two percentile points of the loss probabilities we introduce when losses are symmetric.

Not surprisingly, both NALD and landmark pings perform poorly when we inject asymmetric losses in the neighbor WiFi segment. In this scenario, NALD infers a negative loss probability on dc. In practice, the inference of negative probabilities of loss indicate the violation of the assumption that losses are symmetric on the WiFi segment. When injecting asymmetric losses in the access link NALD properly infers the correct loss probability with small error. Landmark pings, however, cannot pinpoint losses occurring on the uplink versus the downlink.

6.3.2 Real-world experiments
indicate asymmetric losses in the WiFi or in the LAN. An
with the controlled experiments, negative loss probabilities
in 89 rounds (74% of rounds with losses). As we have seen
losses. Finally, NALD infers negative probabilities of losses
suffers from losses less often than access links. WiFi retrans-
fact that access uplinks often have lower capacity than the
access uplink than on the downlink. This result reflects the
the five homes we measure experience more losses on the
rounds (53% of rounds with losses). Figure 7 shows that
infers losses larger than 1% in at least one link in only 64
results from the figure. In general, losses are low. NALD
serve losses in the LAN in four rounds, so we omit these
of the loss probabilities we infer for

\[ \hat{P} \]

The probabilities of loss
\[ \hat{P}_{abcdcb} = 0 \]
because SFR WiFi blocks ICMP. The probabilities of loss
\[ \hat{P}_{fcedcb} \] and \[ \hat{P}_{abcdcb} \] are the most
correlated with correlation coefficient of 0.48. This coeffi-
cient corresponds to a moderate correlation. The correlation
coefficients of all other pairs of network segments were low.
This result indicates that it is often safe to assume that losses
are independent.

**Losses in practice** NALD finds a solution to the systems
of equations in 364 rounds, but we only detect losses in 120
rounds. Figure 7 presents the cumulative density function of
the loss probabilities we infer for \( cd, dc \), and \( ef = fe \)
in the 120 rounds where we detect some loss. We only ob-
serve losses in the LAN in four rounds, so we omit these
results from the figure. In general, losses are low. NALD
infers losses larger than 1% in at least one link in only 64
rounds (53% of rounds with losses). Figure 7 shows that
the five homes we measure experience more losses on the
access uplink than on the downlink. This result reflects the
fact that access uplinks often have lower capacity than the
downlink. We also find that the link to the WiFi neighbor
suffers from losses less often than access links. WiFi retrans-
misions at the link layer and rate adaption mitigate packet
losses. Finally, NALD infers negative probabilities of losses
in 89 rounds (74% of rounds with losses). As we have seen
with the controlled experiments, negative loss probabilities
indicate asymmetric losses in the WiFi or in the LAN. An
explanation is that the topology from Figure 3 is not a good
model of the FreeWiFi implementation. FreeWiFi tunnels
traffic from the end-host inside the core network of FreeWiFi
(i.e., the first IP hop appears after the neighbor’s access link,
which is asymmetric). Although this result is a limitation of
NALD in today’s network, NALD technique remains valid
for other WiFi community implementations.

7. DEPLOYMENT SCENARIOS

The deployment of NADD/NALD requires a couple of
prerequisites. First, to infer round-trip delays to the gate-
ways, these gateways have to respond to any flavor of ping
(e.g., ICMP echo requests, TCP-SYN handshake). Second,
to enable cycle probes, we have to circumvent the NATs im-
plemented in today’s gateways, which requires gateways to
enable UPnP. Third, the end-hosts under test must be multi-
homed. Our analysis in Section 2 shows that more than half
of the homes we measured in France have a connection to
WiFi neighbors through WiFi communities. In addition, our
analysis of HomeNet Profiler data confirms that gateways to-
day do respond to pings and that about a third of gateways
support opening ports with UPnP. These results suggest that
expert end-users in particular in urban areas can already de-
ploy NADD/NALD in some homes today.

In the future, we envision that neighbor-assisted diagnosis
will mainly be deployed by ISPs, over-the-top (OTT) service
providers, or will be embedded in diagnosis applications.

One major concern for ISPs is their ability to diagnose
problems of their customers. The more ISPs know about
where problems occur, the better they will be able to trou-
bleshoot these problems and decrease the costs of the cus-
tomer support centers. With increasingly complex home-
network configurations and with an increasing number of
connected devices in the homes, the knowledge of where
losses and delays occur is of major importance. Some ISPs
are tackling this problem by deploying additional hardware
(as for example with the boxes deployed with SamKnows) to
have a vantage point in the homes. Many ISPs also deploy
the Set-top-boxes, which they could use as a measurement
vantage point in the customer network. The ISP have all
the ingredients for a successful and controlled deployment
of NADD and NALD. First, ISPs control the home gateway,
so they can configure gateways to allow NADD and NALD
probes. Second, ISPs manage their WiFi community and can
designate WiFi credentials NADD and NALD require.

Over the last few months, we have pitched the idea of
neighbor-assisted home network diagnosis to multiple Eu-
ropean access providers and gained a lot of interest in such a
solution. They see the opportunity to simplify troubleshoot-
ing by providing detailed information about where delays
and losses occur. Moreover, they all already control one or
multiple devices in the end user homes. They are also inter-
ested in the possibility of implementing NADD and NALD
directly on the gateway. The gateway is well positioned be-
tween the ISP realm and the home environment and has vis-

![Figure 7: Inferred loss probabilities, we show 120 rounds with losses](image-url)
ibility to both worlds. Of course, one would have to adopt the equations accordingly, but the gateway remains a very interesting target.

Besides ISPs, over-the-top service providers are looking into the issue of understanding the network issues from a customer perspective. Most of the OTT services available today (such as Apple TV, Roku, Amazon FireTV, and alike) are deployed with boxes that are often connected over Ethernet. OTT service providers could then run NADD/NALD to determine whether delivery errors are coming from the home network or from the access network. So complaints could be better managed and reported more transparently to the customers. If an OTT service is popular enough, it could run its own WiFi community to perform neighbor-assisted diagnosis. Furthermore, the multi-homing of the OTT boxes could even be used to improve data delivery using multiple network paths.

8. RELATED WORK

Techniques to measure and diagnose Internet performance have a long history. For example, tools such as the Network Diagnosis Tool [5], Network Path and Application Diagnostics [22], Tulip [19], and Netalyzr [18] send active probes to diagnose end-to-end path performance issues. Different from NADD and NALD, these tools do not pinpoint whether network delays/losses occur in the home or the access network; they do not identify the direction in which delays/losses occur either. Sting [26] identifies the direction of end-to-end losses, but not the location of network losses.

More recently, there has been a surge in tools to measure residential broadband access performance. Researchers have developed methods to measure the performance of a large number of access links from servers connected to the Internet [8, 12]. These methods are useful to get a large-scale view of residential access performance, but not to diagnose performance bottlenecks of any given home. Because the measurements are conducted from outside the home network, they cannot capture performance bottlenecks inside the home network. Services like speedtest.net and grenouille.com measure the performance of access ISP networks from an end-host connected to the home network. The end-to-end performance these tools report does capture performance bottlenecks in the home network, if they exist. These tools, however, do not distinguish whether the bottleneck they measure is in the access or the home network. To explicitly measure access network performance, Sundaresan et al. [29] deploy active measurement tools directly on the home gateway. Their work focus on characterizing access network performance and not on pinpointing whether performance bottlenecks originate in the home or in the access network. In our future work, we plan to adapt NADD and NALD to work from home gateways.

A novel aspect of neighbor-assisted network diagnosis is to use multi-homing (in particular, WiFi neighbor and community networks) for the purpose of delay and loss diagnosis. Goma et al. [16] use the idea of a home user attaching to a neighbor’s access point to save energy and to use bandwidth from neighboring WiFi access points. The equation systems underlying NADD and NALD are a typical example of network tomography [4, 6, 15, 32]. The complexity of applying network tomography in practice comes from the need to coordinate measurements from different hosts. NADD and NALD avoid these issues because they work with the control of a single, multi-homed vantage point. As a result, our techniques work with no coordination or clock synchronization among different machines. This feature makes implementations of NADD and NALD practical, as our proof of concept deployment in five homes in Paris shows. At coarser time scales, a number of systems correlate measurements from a many vantage points to infer the location of performance degradation [7, 21, 23, 25]. These systems techniques could leverage neighbor-assisted measurements to increase the number of vantage points and also share NADD and NALD results.

9. CONCLUSION AND FUTURE WORK

This paper introduces neighbor-assisted network diagnosis and designs and evaluates two techniques based on this idea: NADD and NALD. These techniques use both the home network and a neighboring wireless network to send network probes, ranging from simple RTT ping measurements to more complex cyclic probes, to determine the uplink and downlink latency and losses in the home and access networks. We validate our techniques with controlled experiments and also evaluate NADD and NALD in the wild with a proof-of-concept deployment. We show that our techniques are accurate when the LAN link and the WiFi link both have symmetric delays and losses. In practice, delay asymmetry in the WiFi link may harm NADD’s accuracy. If asymmetric delays occur, one can compensate them by incorporating modulation rates of sent and received frames into the delay computations. To avoid measurements errors due to congested home networks, the end-hosts could monitor WiFi frames from other clients to detect congestion at the access point, like EZ-flow does for mesh-networks [2].

Our proof-of-concept deployment leverages the availability of neighboring community WiFi networks. However, WiFi communities policies often limit the types of probes hosts can send. As we discussed in Section 7, a more interesting scenario is when an ISP runs a tool like NADD or NALD for troubleshooting purposes. In that case, the ISP will allow measurement hosts to send all the required measurement probes and use the measurement results to debug and troubleshoot their network.

In this paper, we considered only the case of a single neighbor network. In a future evolution of our solution, we will examine how to generalize our equation systems when more than one extra path is available. We also plan to explore the multi-dimensional tradeoff between overhead (due to packet train length, the number of trains per round, and
number of measurement rounds) and accuracy.

Finally, we plan to extend our neighbor-assisted network diagnosis to provide additional performance measurements beyond directional per-hop delays and losses. Neighbor-assisted network diagnosis empowers end-hosts to run reachability and performance tests over a variety of network paths (like NetworkMD [21] does from monitoring traces collected within an ISP network). Periodically monitoring reachability and performance of the connection to the top web sites (for example as reported on Alexa), will help the end-host pinpoint network problems. A single end-host could thus assess whether the same performance or connectivity problem occurs for a single home, one ISP, or if multiple ISP are concerned. Such additional debugging information will help to decrease the number and the duration of customer-service center calls. Further, such measures provide insights of the performance evolution of network service providers.

10. REFERENCES

