



**HAL**  
open science

## Predicting the effect of home Wi-Fi quality on Web QoE

Diego Neves da Hora, Renata Teixeira, Karel van Doorselaer, Koen van Oost

### ► To cite this version:

Diego Neves da Hora, Renata Teixeira, Karel van Doorselaer, Koen van Oost. Predicting the effect of home Wi-Fi quality on Web QoE. SIGCOMM Workshop on QoE-based Analysis and Management of Data Communication Networks (Internet-QoE), ACM, Aug 2016, Florianópolis, Brazil. 10.1145/2940136.2940146 . hal-01339522

**HAL Id: hal-01339522**

**<https://inria.hal.science/hal-01339522>**

Submitted on 5 Sep 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Predicting the effect of home Wi-Fi quality on Web QoE

Diego da Hora<sup>1,2</sup>, Renata Teixeira<sup>1</sup>  
<sup>1</sup>Inria  
{First.Lastname}@inria.fr

Karel Van Doorselaer<sup>2</sup>, Koen Van Oost<sup>2</sup>  
<sup>2</sup>Technicolor  
{First.Lastname}@Technicolor.com

## ABSTRACT

Wi-Fi is the preferred way of accessing the Internet for many devices at home, but it is vulnerable to performance problems. The analysis of Wi-Fi quality metrics such as RSSI or PHY rate may indicate a number of problems, but users may not notice many of these problems if they don't degrade the performance of the applications they are using. In this work, we study the effects of the home Wi-Fi quality on Web browsing experience. We instrument a commodity access point (AP) to passively monitor Wi-Fi metrics and study the relationship between Wi-Fi metrics and Web QoE through controlled experiments in a Wi-Fi testbed. We use support vector regression to build a predictor of Web QoE when given Wi-Fi quality metrics available in most commercial APs. Our validation shows root-mean square errors on MOS predictions of 0.6432 in a controlled environment and of 0.9283 in our lab. We apply our predictor on Wi-Fi metrics collected in the wild from 4,880 APs to shed light on how Wi-Fi quality affects Web QoE in real homes.

## 1. INTRODUCTION

Tablets, laptops, and smartphones almost exclusively use Wi-Fi to connect to the Internet at home. Wi-Fi performance, however, is highly variable. Competing Wi-Fi networks can cause contention and interference while poor channel conditions between the station and the access point (AP) can cause frame losses and low bandwidth. In some cases, the home Wi-Fi network can bottleneck Internet access [15]. While problems in the Wi-Fi network may affect several network Quality-of-Service (QoS) metrics, such as latency and bandwidth, users will typically only notice a problem when poor Wi-Fi affects the Quality-of-Experience (QoE) of Internet applications. For example, a Wi-Fi network with low bandwidth may go unnoticed unless the time to load Web pages increases significantly. A user observing degraded QoE due to Wi-Fi problems may mistakenly assume there is a problem with the Internet Service Provider (ISP) network. Our discussions with residential ISPs confirm that often customers call to complain about problems in the home Wi-Fi and not the ISP

network.

Although there is a large body of work on diagnosing Wi-Fi pathologies, prior work has focused on QoS metrics with no attempt to identify when Wi-Fi quality affects QoE. The general problem of mapping network QoS to QoE has been studied for some applications (e.g., on-line video [1, 2], Web browsing [6], or Skype [16]). Yet, there is little work bridging the gap between Wi-Fi metrics and QoE. This paper is a first step to address this gap.

Our end goal is to build a system for ISPs to predict when home Wi-Fi quality degrades QoE. ISPs can use this system to detect customers experiencing poor QoE to proactively trigger Wi-Fi troubleshooting. ISPs often control the home AP, so we leverage Wi-Fi metrics that are available on commercial APs. Detecting when Wi-Fi quality degrades QoE using these metrics is challenging. First, we have no information about the applications customers are running at any given time. ISPs avoid capturing per-packet traffic traces from customers, because of privacy considerations and the overload of per-packet capture. Thus, we must estimate the effect of Wi-Fi quality on QoE of popular applications, which most customers are likely to run. In this paper, we study Web as a proof of concept, as a large fraction of home traffic corresponds to Web [3, 10]. Second, application QoE may be degraded by factors other than the Wi-Fi quality (e.g., poor Internet performance or an overloaded server). Although a general system to explain any QoE degradation would be extremely helpful, our monitoring at the AP prevents us from having the end-to-end view necessary for such general task. Instead, we focus on identifying when Wi-Fi quality degrades QoE. Finally, Wi-Fi metrics available in APs are coarse aggregates such as the average PHY rate or the fraction of busy times. It is unclear how to map these coarse metrics into QoE.

This paper uses regression analysis to build a predictor that given Wi-Fi metrics measured at the AP estimates Web QoE (§3). We generate training samples in a controlled environment, where we ensure that Wi-Fi quality bottlenecks Web QoE in a diverse set of Wi-Fi conditions (§2.1). Instead of conducting tests with real users, we measure Web page load times and use

the standard ITU-T G.1030 model [8] to obtain a mean opinion score (MOS) in the range [1-5].

We validate this predictor using nested cross-validation, and MOS predictions show a root-mean square error of 0.6432. Our tests with data collected in our lab (with real Wi-Fi conditions) where we collect both Wi-Fi metrics and Web page load times over a two weeks period also show a low root-mean square error of 0.9283 (§4).

We apply this predictor on Wi-Fi metrics collected from 4,880 home customers of a large Asian-Pacific ISP over a period of 40 days (§5). This analysis shows that although Wi-Fi quality in the wild is often good for light Web pages, when Web pages are more complex Wi-Fi quality degrades Web QoE around 20% of time.

## 2. TRAINING SET

This section builds the training set for predicting the effect of home Wi-Fi quality on Web QoE. To generate the training set, we need instances where we know both Wi-Fi metrics and the target QoE metric, which we generate with controlled experiments.

### 2.1 Experimental setup

We emulate a home Wi-Fi network on the testbed shown in Figure 1, where we carry out automated Web browsing tasks. We use a Technicolor AP with a Broadcom BCM6362 NIC with 802.11n 2x2 technology, and a MacBook pro station (STA) with 802.11n 2x2 technology. We connect to the Internet through an Ubuntu 12 gateway using TC to limit download / upload bandwidth to 16 / 1 Mbps, emulating popular ADSL offers both in France and Belgium (we add neither extra delay nor packet loss).

In each scenario, we use PhantomJS, a headless browser, to sequentially access 10 times a set of 10 pages, while we monitor Wi-Fi, network, and application metrics. We chose a mix of pages figuring in the Alexa TOP 20 pages in Belgium. We accessed the front pages of the following domains: bing.com, google.be, twitter.com, live.com, wikipedia.org, facebook.com, yahoo.com, amazon.fr, nieuwsblad.be, and hln.be.

We vary the Wi-Fi conditions over two axes:

*Link quality.* When the link quality between AP and STA is good, throughput is high. When the link quality is poor, however, the rate adaptation algorithm uses slow but robust modulation rates (or PHY rates) to ensure the reception of frames. We vary the link quality by introducing attenuation in the path between AP and STA using a programmable attenuator. We introduce 6, 12, 15, 18, 19 and 20 dBs of attenuation.

*Medium availability.* We focus on the medium availability of the AP, since most of the traffic flows from the AP to the STA. The IEEE 802.11 medium access protocol establishes that nodes only transmit after sensing the medium idle. The medium can be busy due to the interference of Wi-Fi or non-Wi-Fi sources, thus we execute two sets of experiments.

Table 1: PLT to MOS mapping per page.

Web page	MOS=5	MOS=3	MOS=1
bing.com	0.287 s	1.694 s	10 s
google.be	0.497 s	2.228 s	10 s
live.com	1.689 s	4.109 s	10 s
twitter.com	1.723 s	4.150 s	10 s
wikipedia.org	1.871 s	4.325 s	10 s
yahoo.com	2.518 s	5.018 s	10 s
facebook.com	2.849 s	5.336 s	10 s
nieuwsblad.be	3.855 s	6.208 s	10 s
amazon.fr	3.998 s	6.322 s	10 s
hln.be	4.176 s	6.462 s	10 s

On the first set of experiments, we evaluate the impact of non-Wi-Fi interference on Web browsing. We use the signal generator to inject a narrowband sinewave to block the AP’s clear channel assessment (CCA). The signal generator is an arbitrary waveform generator, capable of handling swept sinewave transmissions. We sweep the interfering sinewave in and out of the spectral CCA range of the AP every 200ms, as illustrated in Figure 2. We control the medium business by varying the percentage of time the sinewave is inside and outside the CCA range, creating scenarios with medium availability of: 100%, 75%, 50%, 25%, and 12.5%. We verified that Wi-Fi counters on the AP observe the intended medium availability with error below 5%.

On the second set of experiments, we use a second pair of AP / STA to generate competing Wi-Fi traffic blocking the medium, as shown in Figure 1. An Ubuntu 12 computer, connected through a gigabit interface to the AP generates UDP traffic to an android tablet with an 802.11n 1x1 NIC executing iperf in server mode. We configure the interfering link to only use PHY rate 5.5 Mbps, to hold the medium for longer per medium access. The interfering traffic determines the medium availability when the link under test is not used. We found that by generating constant bit rate UDP traffic of 0.7 Mbps, 1.1 Mbps, 2.1 Mbps and 2.9 Mbps we obtain, respectively, 70%, 50%, 30% and 15% medium availability, measured by the Wi-Fi counters on the AP.

To guarantee that any degradation is due to Wi-Fi problems, we examine the WAN RTT, which captures network and server processing delay. We filter out any samples where the median WAN RTT is above a threshold  $t$ , for each page. For each page, the threshold is given by the average 90<sup>th</sup> percentile of the RTT during baseline experiments (i.e., perfect Wi-Fi conditions).

### 2.2 Estimating Web QoE

Measuring QoE requires tests with real users, but it would be hard to convince people to conduct enough tests to sweep all the combinations of parameters we outlined in the previous section. Instead, we use the page load time (PLT) as the main indicator of Web browsing experience. We measure the application *PLT* using the *onLoad* event, triggered when the browser finishes loading the Web page and attached objects (e.g.,

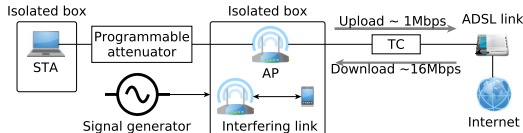


Figure 1: Components of the Wi-Fi testbed

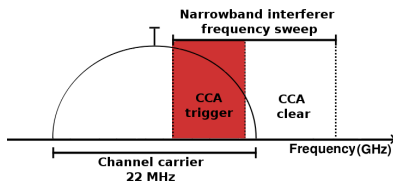


Figure 2: Frequency-swept medium business emulation

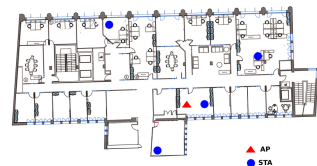


Figure 3: AP / STA positions on office environment

javascript code, images).

We model QoE from PLT as follows. The ITU-T G 1030 maps the browsing session time to a MOS score in the range [1-5] [8]. It shows a logarithmic relationship between PLT and MOS, which is accepted to be true for both task driven as well as one page browsing sessions [6]. We use ITU-T G 1030 model to map PLT into MOS providing two parameters: the *maximum PLT*, which maps to the minimum QoE score (1); and the *minimum PLT*, which maps to the maximum QoE score (5). We set the maximum PLT to 10 seconds for all pages, since users tend to lose focus after 10 seconds of waiting time [6]. Given that different pages have different PLTs on the baseline scenario, we set the model’s minimum PLT to the mean PLT per page during the baseline scenario. Our reasoning for this choice is to better match user’s expectations. When users visit a complex page they know it will be slower than visiting a simple page like Google. Table 1 shows the chosen PLTs to the QoE scores of 5, 3 and 1 for the ten pages we measure.

*Model limitations* Other influence factors not covered by this model may affect QoE, such as context, page content and aesthetics [18], and page loading behavior [11]. We only consider factors that are likely to influence the relationship between Wi-Fi quality and Web QoE. Similarly to how network QoS metrics is farther from QoE than application quality metrics [11], Wi-Fi metrics should be farther from QoE than network QoS metrics. Therefore, we do not expect a high accuracy model between Wi-Fi metrics and QoE, but rather a model that allow us to detect when Wi-Fi quality significantly degrades Web QoE. The QoE of rich internet applications (e.g. AJAX, WebRTC apps) is not covered by this model and is left for future work.

### 2.3 Training samples

For each page visit in our controlled experiments, we generate a sample for our training set with Wi-Fi metrics and the estimated Web QoE. We passively collect the set of Wi-Fi metrics shown in Table 2 from the AP. The RSSI and average Tx / Rx PHY rate should strongly correlate with variations in the SNR, since they all indirectly measure the link quality. The percentage of time the medium is busy due to Wi-Fi and non Wi-Fi traffic help identify variations in the medium availability. Frame delivery ratio counters should strongly correlate with high frame loss scenarios. We build each sample by aggregating the Wi-Fi metrics over a period

Table 2: Wi-Fi metrics passively collected.

Feature	Description
AvgTxPHY	Avg. PHY rate of frames sent
AvgRxPHY	Avg. PHY rate of frames received
FDR	Frame delivery ratio to STA
RSSI	Received signal strength indicator
BUSY	% of time the medium is busy
WiFi	% busy due to Wi-Fi traffic
nonWiFi	% busy due to non-Wi-Fi signal

of 30 seconds starting at the Web page access time, and we label each sample with the estimated Web QoE.

## 3. WEB QOE PREDICTOR

This section uses the training set to build a predictor of Web QoE from Wi-Fi metrics. We cast the problem of predicting the QoE as a regression problem. Given the set of Wi-Fi metrics, shown in Table 2, we want to predict the estimated Web QoE.

### 3.1 Prediction algorithm

We evaluate four well-known prediction algorithms: support vector regressor (SVR), decision tree regressor (DTR), linear regressor (linear), and gaussian naive bayes (GNB). Since SVR requires feature normalization, we manually normalize the features so that values fall into the range  $[-1 : 1]$ . We use grid-optimization to choose the best combination of  $\lambda$ ,  $\gamma$ , and  $\epsilon$  parameters for SVR, and maximum depth and minimum number of samples per leaf for DTR.

Since the baseline PLT of different Web pages vary by more than one order of magnitude (Table 1), we split the dataset per visited page and hence build one predictor per page. We evaluate the prediction accuracy by comparing root-mean square errors (RMSE) obtained through 10-fold cross-validation for linear and GNB and 10-fold nested cross-validation for SVR and DTR, to obtain unbiased accuracy estimations [17]. Figure 4 present the results. For comparison, we show also the accuracy of a constant regressor, which returns the average MOS of the dataset. We see that both SVR and DTR are reasonably accurate, whereas linear and GNB perform significantly worst. We opt to use the SVR predictor, as it is the most accurate.

To understand the influence of the visited page on the estimated QoE, we show the decision curves of SVR predictors using two features for three different pages that vary significantly in terms of complexity [4]: google.be representing a light page (Figure 7b), facebook.com an

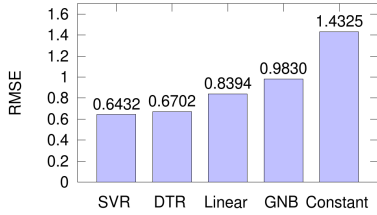


Figure 4: Predictor’s accuracy

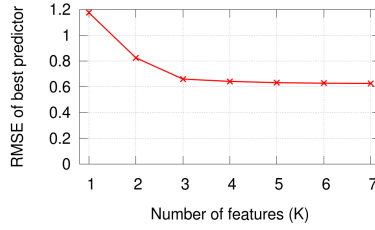


Figure 5: Effect of number of features on SVR accuracy

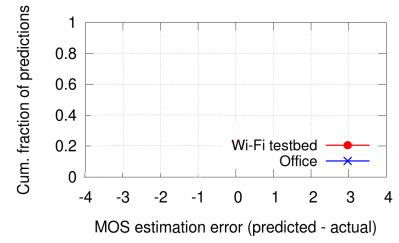


Figure 6: Prediction errors in Wi-Fi testbed and office

average page (Figure 7c), and amazon.fr a complex page (Figure 7d). The SVR model learns similar boundary conditions on all different pages. When Wi-Fi conditions are perfect, with BUSY near 0 and AvgTxPHY rate at maximum, we observe a predicted MOS of 5 for all pages. Similarly, when either AvgTxPHY is close to 0 or BUSY close to 100%, we observe a predicted MOS of 1. However, the requirements for  $MOS > 3$  significantly differs per visited page, with complex pages requiring much better Wi-Fi conditions than light pages.

### 3.2 Feature selection

To understand which set of features is the most relevant for the prediction task, we use the following feature selection method. We choose a fixed number of features  $K$ , and train SVR predictors using all possible combinations of  $K$  features. Then we select the best SVR predictor using  $K$  features. Figure 5 shows the RMSE of the best SVR predictor varying  $K$  from 1 to 7. The best predictor with  $K = 1$  uses FDR, with  $K = 2$  uses BUSY and AvgTxPHY, and with  $K = 3$  uses these three features. Adding a fourth feature is unnecessary as RMSE remains at  $\approx 0.6$  as we add more features. This set of selected features is intuitive. FDR is a general Wi-Fi quality indicator, whereas BUSY and AvgTxPHY points to specific kinds of Wi-Fi impairments. This result also confirms that RSSI is not a good indicator of Wi-Fi quality [13]; AvgTxPHY is a better alternative.

## 4. VALIDATION

The previous section evaluates prediction accuracy using cross validation, now we evaluate the predictor with a test set collected in a more realistic environment. We create the office dataset by conducting the same set of experiments in our lab in Paris, where there is real Wi-Fi and non Wi-Fi interference. We vary the position of STA as shown in Figure 3. We execute Web browsing task as described in §2.1 every 30 minutes, from 6am to 11pm. Medium availability and link quality vary in an uncontrolled way, with measured medium availability varying from 90% to 30%.

We run these experiments for two weeks and generate test samples as described in §2.3. The office dataset, however, contains many similar samples, because in the early morning and late night there’s very little Wi-Fi and non-WiFi interference. We solve this by over-sampling the dataset. We discretize each feature into

ten possible values, and bin samples based on the discretized feature space, with similar samples belonging to the same bin. Then, we prune bins holding less than 0.5% of the set size to avoid over-representing them. Finally, we over-sample each bin until all bins have the same number of samples.

Figure 6 shows MOS estimation errors of the Wi-Fi testbed dataset, obtained through 10-folded cross-validation, and the MOS estimations errors on the office dataset. We see that the predictor presents 90% and 85% of estimations with error below 1 MOS point in the Wi-Fi testbed and office datasets, respectively. This is good evidence that the predictor learns a robust model on the Wi-Fi testbed. The RMSE of the predictor was 0.9283 on the office dataset, compared to 0.6432 on the Wi-Fi testbed dataset.

## 5. EFFECT OF WI-FI QUALITY ON WEB QOE IN THE WILD

We leverage measurements of an existing deployment to assess the effects of Wi-Fi quality on Web QoE in the wild. We measure Wi-Fi metrics from APs deployed in 4,880 residential customers of a large Asian-Pacific ISP, reporting over 23 thousand devices to a backend server. We calculate the Wi-Fi features from Table 2 and store one sample per device every 30 seconds. We analyze data collected from February 18 to March 30, 2016, which gives us a total of 180 million samples.

Unfortunately, this deployment didn’t collect FDR. Therefore, we build an SVR predictor using AvgTxPHY, Wi-Fi, and nonWi-Fi, instead. We use  $\gamma$ ,  $\lambda$ , and  $\epsilon$  parameters found in §3.1. This predictor presents a RMSE of 0.7924 through cross-validation. To account for page complexity [4], we build three predictors: *light*, representing very lightweight pages such as search engines front pages, we use samples of page loads of google.be for building this predictor; *complex*, representing pages with many objects / images (amazon.fr in our case); and *average*, representing pages of intermediary complexity (facebook.com).

Figure 8 shows the distribution of predicted MOS for the three predictors in the deployment dataset. We see that over 60% of samples present MOS above 4.8 for all predictors. This result is reassuring as in most cases home Wi-Fi quality is good for Web. We consider a *poor Web QoE* instance when the predictor estimates MOS below 3, where perceived QoE degradation ranges from

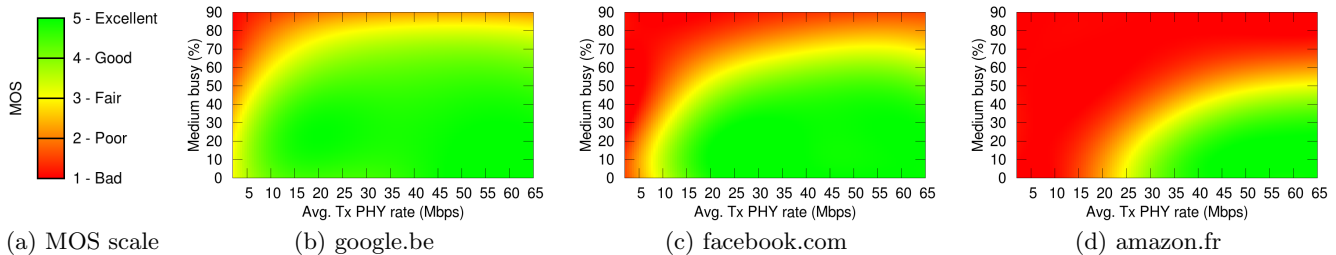


Figure 7: MOS scale and visualization of SVR models

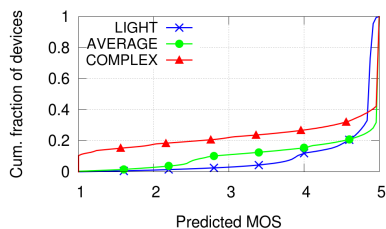


Figure 8: Predicted MOS frequency in the wild

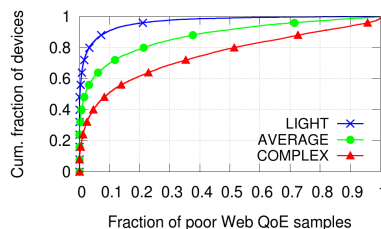


Figure 9: Fraction of poor Web QoE episodes per device

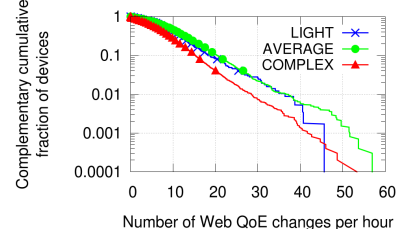


Figure 10: QoE changes for devices with poor Web QoE fraction  $\geq 0.2$

“slightly annoying” to “very annoying” [9]. The complex predictor estimates that Wi-Fi quality causes poor Web QoE in 22% of samples, whereas the average predictor estimates poor Web QoE in 11% of samples. We observe that Wi-Fi quality rarely degrades Web QoE for light pages, as expected from Figure 7b.

We now focus on poor Web QoE events per device. We select one day of data from the deployment, and analyze over 7.5 million samples from 4,880 APs to obtain 13 thousands devices connected for at least one hour. Figure 9 shows the fraction of poor Web QoE samples per device using light, average, and complex predictors. We see that 4.5%, 21%, and 49% of devices have more than 20% of poor Web QoE samples for light, average, and complex pages respectively.

We look at devices showing Wi-Fi problems to see if poor Web QoE episodes are consistent or intermittent. For each device with at least 20% poor Web QoE samples, we count instances when predicted Web QoE changes from  $MOS < 3$  to  $MOS > 3$  (and vice-versa). Figure 10 shows the number of Web QoE changes per hour. We see that 54% of devices present less than 10 Web QoE changes per hour, considering the average predictor. In these cases, Wi-Fi problems can be detected and tackled on the spot, either by a help desk operator or an end user equipped with a proper application, by executing changes in the home Wi-Fi network and immediately evaluating its effects. However, 5% of devices show more than 25 Web QoE changes per hour. Solving problems in these Wi-Fi networks is challenging since changes cannot be evaluated immediately but require a long term monitoring approach to ensure that the applied changes were effective in solving the observed QoE issues.

## 6. RELATED WORK

Several papers have studied Wi-Fi performance in the

wild by instrumenting APs to collect QoS metrics. Pei et al. [15] studies the latency in the Wi-Fi hop using 47 APs deployed in a university campus. Ioannis et al. [14] obtain insights from the Wi-Fi performance using passive measurements from 167 homes on a period of four months. Biswas et al. [3] characterizes Wi-Fi network usage and performance on over 20 thousand industrial networks. Each of these studies adds to our understanding of Wi-Fi quality in different settings, but none of them addresses the issue of how Wi-Fi quality affects QoE as we do here.

Quality of experience has received a lot of attention. We identify two main quality assessment methodologies. Subjective quality assessment studies use human subjects to explicitly evaluate user experience [6]. Alternatively, objective quality assessment studies evaluate user experience using quality metrics such as SSIM for image and VQM for video [1]. Recent studies infer user experience from engagement centric metrics [2]. We avoid costly involvement of human subjects by mapping application quality metrics into subjective quality metrics, similar to Hohlfeld et al. [7].

There is also work on mapping network QoS to QoE for different applications, such as UDP and TCP video streaming [5, 19], VoIP [12], and Skype [16]. All these previous studies focus on mapping end-to-end network metrics and not on establishing the relationship between Wi-Fi metrics and Web QoE as we do here.

## 7. CONCLUSION

This paper is the first to shed light on the effects of home Wi-Fi quality on Web QoE. We create a predictor of Web QoE based solely on Wi-Fi metrics commonly available in commercial APs. To generate the training set, we execute web browsing tasks on a diverse set of scenarios where Wi-Fi bottlenecks the Web QoE. We estimate Web QoE using the standard ITU-T G.1030 QoE

model, mapping PLT to MOS. Our evaluation of different prediction algorithms shows that SVR is the most accurate for predicting Web QoE from Wi-Fi metrics. We obtain an RMSE of 0.6432 with cross-validation with samples from the controlled setting and of 0.9283 when testing with samples taken in our lab. Finally, we apply our predictor on Wi-Fi metrics collected in the wild from 4,880 APs over a period of 40 days. We find that Wi-Fi quality is mostly good for Web—in more than 60% of samples Wi-Fi quality does not degrade Web QoE. When we consider complex or average Web pages, however, Wi-Fi quality degrades Web QoE in 22% and 11% of samples, respectively. Moreover, we saw that 21% of devices present more than 20% of poor Web QoE samples, with 5% of these showing highly intermittent QoE degradations, which are particularly hard to diagnose. This indicates that we need a long-term monitoring approach to detect and fix problems.

This paper is a first step into understanding the effect home Wi-Fi quality on user experience. Our goal is to build a general predictor of QoE from Wi-Fi quality. For this purpose, we plan to study how Wi-Fi affects other popular applications such as video streaming, VoIP, and online gaming. Finally, we want to perform a characterization of home Wi-Fi quality in the wild, focusing on the user experience.

## Acknowledgements

We thank Luc Gyselink and Nick Godman for helping with the Wi-Fi testbed and deployment data analysis. Thanks also to our shepherd Tobias Hossfeld and the anonymous reviewers for their helpful comments. This work was supported by the European Community Seventh Framework Programme (FP7/2007-2013) no. 611001 (User-Centric Networking).

## 8. REFERENCES

- [1] M. Alreshoodi and J. Woods. Survey on qoe\ qos correlation models for multimedia services. *arXiv preprint arXiv:1306.0221*, 2013.
- [2] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang. Developing a predictive model of quality of experience for internet video. In *ACM SIGCOMM CCR*, 2013.
- [3] S. Biswas, J. Bicket, E. Wong, R. Musaloiu-e, A. Bhartia, and D. Aguayo. Large-scale measurements of wireless network behavior. In *Proc. ACM SIGCOMM*, 2015.
- [4] M. Butkiewicz, H. V. Madhyastha, and V. Sekar. Understanding website complexity: measurements, metrics, and implications. In *Proc. IMC*, 2011.
- [5] P. Casas, M. Seufert, and R. Schatz. Youqmon: A system for on-line monitoring of youtube qoe in operational 3g networks. *ACM SIGMETRICS Performance Evaluation Review*, 2013.
- [6] S. Egger, T. Hossfeld, R. Schatz, and M. Fiedler. Waiting times in quality of experience for web based services. In *QoMEX*. IEEE, 2012.
- [7] O. Hohlfeld, E. Pujol, F. Ciucu, A. Feldmann, and P. Barford. A qoe perspective on sizing network buffers. In *Proc. IMC*. ACM, 2014.
- [8] ITU. Estimating end-to-end performance in ip networks for data applications. *ITU-T Recommendation G.1030*, 2005.
- [9] Methods for subjective determination of transmission quality. *ITU-T Recommendation P.800*, Aug 1996.
- [10] G. Maier, A. Feldmann, V. Paxson, and M. Allman. On dominant characteristics of residential broadband internet traffic. In *Proc. IMC*. ACM, 2009.
- [11] P. Meenan. How fast is your website? *Communications of the ACM*, 2013.
- [12] M. Papadopouli, P. Charonyktakis, M. Plakia, and I. Tsamardinou. On user-centric modular qoe prediction for voip based on machine-learning algorithms. *IEEE TMC*, 2015.
- [13] A. Patro, S. Govindan, and S. Banerjee. Observing home wireless experience through wifi aps. In *Proc. international conference on Mobile computing & networking*. ACM, 2013.
- [14] I. Pefkianakis, H. Lundgren, A. Soule, J. Chandrashekar, P. Le Guyadec, C. Diot, M. May, K. Van Doorselaer, and K. Van Oost. Characterizing home wireless performance: The gateway view. In *Proc. IEEE INFOCOM*, 2015.
- [15] C. Pei, Y. Zhao, G. Chen, R. Tang, Y. Meng, M. Ma, K. Ling, and D. Pei. Wifi can be the weakestlink of round trip network latency in the wild. In *Proc. IEEE INFOCOM*, 2016.
- [16] T. Spetebroot, S. Afra, N. Aguilera, D. Saucez, and C. Barakat. From network-level measurements to expected quality of experience: the skype use case. In *M & N workshop*, 2015.
- [17] I. Tsamardinou, A. Rakhshani, and V. Lagani. Performance-estimation properties of cross-validation-based protocols with simultaneous hyper-parameter optimization. *International Journal on Artificial Intelligence Tools*, 2015.
- [18] M. Varela, L. Skorin-Kapov, T. Maki, and T. Hossfeld. Qoe in the web: A dance of design and performance. In *QoMEX*, pages 1–7. IEEE, 2015.
- [19] T. Wang, A. Pervez, and H. Zou. Vqm-based qos/qoe mapping for streaming video. In *Proc. IC-BNMT*. IEEE, 2010.