Netradar - Measuring the Wireless World

Sebastian Sonntag, Jukka Manner, Lennart Schulte
Aalto University, Finland
Email: firstname.lastname@aalto.fi

***Abstract*—There exists a number of network measurement tools and web-based services for end users. Almost all of them focus on network bandwidth and possibly latency, and simply report the results to the user. This paper presents Netradar, a somewhat different measurement service. We analyze network bandwidth and latency but also over ten other parameters. We dwell into the data and show various coverage maps and statistics on our web site for the benefit of the end users. Our long term goal is to be able to answer the simple question: why did I get the indicated result at a given time and place?**

I. Introduction

Network access, whether fixed or mobile, has become an absolutely critical service of our society. Our modern way of life, the services offered by the government, cities, employers and business in general are fully networked. We seem to rely on network connectivity as deeply as we rely on water, heating and electricity. With the strong move towards mobile services, smart phones and tablets, a huge part of our daily life is wireless. The industry is competing heavily for the customers, promoting new services and better equipment, and constantly increasing bandwidths and capacity. Looking deeper into this development, one might ask what is the reality in terms of coverage, service quality and stability, and e.g. the effect of the end devices on the perceived Quality of Experience?

There exists tens of different systems, tools and services to analyze network connectivity. They are all built with some unknown background and reason, mostly by a company, and the data is either limited in nature or not available for us. As scientists, we wanted to acquire thorough and scientifically reliable data about mobile networks and devices. As no system able to fulfill our needs was available, we decided to build a system that is based on the best practises of our community. The data available from the database must be as reliable and objective as humanly possible taking into account the nature of mobile and wireless communication.

We started the work in early 2011 and launched the first public version of our Netradar service in June 2012 in Finland, named Nettitutka (Finnish for Netradar). Nettitutka does measurements to a single point in the Finnish high-speed scientific university network FUNET. The service has been hugely popular with a stable user base, and we decided very quickly to develop a scalable system and deploy it into the cloud for the whole world to measure and share information about mobile networks and devices—this system is *Netradar*.

Our system differs from other similar offerings in two major aspects. First, we store a number of different parameters and perform a serious of network-related measurements, not simply network speed and latency as most other platforms focus on. Secondly, our long-term aim is to provide answers to the question "*Why did the user get a certain result?*". All platforms available today focus on reporting values, but we want to take network measurements and analysis much further and seek to understand the reasons. Mobile network technologies, operator configurations and general business strategies, and end user devices all play a role in the outcome of a single measurement run; we aim to open up this complex system. We are in the process of designing algorithms and data base searches to provide answers to end users and researchers.

In this paper, we give an overview of our system and present as case study a series of analysis on the behavior and availability of network bandwidth in the Finnish mobile networks. We show how different factors affect and limit the mobile device downstream bandwidth. There are earlier studies about mobile bandwidth variation but they leave the reasons unanswered [6] [7] [11] [8]. We have identified five major factors: radio technology, coverage, congestion caused by other users, mobile phone itself and handovers. These factors cannot explain the received bandwidth completely, thus there is still room to understand the magnitude and sources for seemingly randomn events. In this study we show the cause for the received bandwidth and analyze the size of impact with different factors. This information can be used to optimize the solutions, such as the buffers, for mobile communications [2].

The rest of the paper is structered as follows. We shortly present various network measurement services in Section 2, followed by a description of Netradar in Section 3. Section 4 presents some interesting analysis related to network bandwidth and in Section 5 we discuss the results and various issues around network measurements. Section 6 concludes our paper.

II. Measurement platforms

Various mobile application stores have actually tens of different measurement tools available, ranging from bandwidth and latencies to radio frequency analysis. In this section, we present and categorize some of the potentially more well-known free measurement systems and software available. In general, we can divide the systems and tools into three categories:

1) Generic bandwidth and latency measurements,
2) Mobile-specific systems, and
3) Other analyzers.

The first group includes tools such as Speedtest.net, SamKnows, Google MLabs, Cisco Global Internet Speed Test GIST, and Bredbandskollen (only swedish). They all report mostly bandwidth numbers and sometimes latency. The MLabs also provides raw data from those measurements.

The second group includes tools such as mySpeedTest, Opensignal, Net.isfaction, Mobispeed, Rootmetrics, 3GTest [4] and QIP Speed Test (only russian). These have more focus on mobile bandwidth or signal strength measurements. Raw data is not available.

In the last group we include services like NetAlyzr [5], that analyzes the technical features of the host's IP access, or Vellamo that does mobile web browser benchmarking among other tests. There is also a more scientific NetPiculet for analyzing middle-boxes [10]. These tools are not in our primary focus.

The common goal with the first two groups of tools is that they simply do some targeted measurements and show the user the result. Any associated web service also focuses on reporting but fail to provide more information about the reasons for the observed and measured performance numbers. Our system and service first of all seeks to combine all possible data about the mobile connectivity, not simply one banwidth value, and secondly we aim to analyze the data deeply and provide statistics and other analysis to the end users, and consequently to the scientific community. Moreover, existing platforms seem to focus on first-hop link maximum bandwidth, while we believe some notion of sustainable bandwidth is much more meaningful.

III. Platform

In this section we explain our measurement system called Netradar. We start with an overview, followed by a presentation of the values and parameters used in this study.

A. Overview

Netradar is a client-server system, a mobile application for measuring network quality and a server setup to measure against. The infrastructure is currently distributed in the cloud over three continents. Measurements are done to the closest server, that models fetching data from a close host or cache.

We have replicated and distributed the database, web servers and measurement servers to multiple servers inside one continent. Additionally we monitor server load in realtime to launch new instances and limit access for better response time. As a last procedure the measurement servers only allow a fixed number of simultaneous clients. This way we seek to maintain the measurement bottlenecks at the client side and thus the results describe the performance of the end user's device and network, not the server side.

We have client applications for Android, iOS, Symbian, Windows Phone, Meego and Maemo; a Java-based application for laptops is also written. The measurement set is identical in every platform. However, there are differences in different platform, e.g. it's not possible to gather signal strength from iOS or Windows Phone devices.

We store the operating system, device type and vendor to allow us compare and analyze the difference in end user systems. For every installation we generate a unique identification that enables us to analyze a single device data. There are currently over 300 different mobile phones and tablets in our database.

There are two possible ways to run measurements. User may launch the measurement by using the UI. The application is also able to go into the background and launch measurements now and then. The interval between measurements is user adjustable, currently from 1 to 120 minutes.

The client stores timestamps at a millisecond resolution at the beginning and the end of the test. Also every event, e.g. handover, and value change, e.g. location, is stored with a timestamp. This allows simulating the whole measurement afterwards. For example, in this study we show differences between day and night in Finland.

B. Bandwidth

Netradar has both download and upload measurements. The client application transfers as fast as possible with TCP for 10 seconds. At this stage, we only use one data transfer at a time due to two primary reasons. Firstly it is clear and easy to analyze the performance on a packet-level when we know that there should not be other competing transfers. Secondly, high speeds are typically used for streaming content or download of big files, both of which are based on a single data flow. In the future, should the need arise, we can implement other measurement methods e.g., packet pair and packet train. Use of multiple flows would give incredible results comparing the bandwidth with real applications.

The connection is made to our measurement server where we only allow a limited number of simultaneously connections. This way we avoid a situation where the measurement infrastructure would be a bottleneck. As LTE deployments and over 100 Mbps wireless speeds become reality, we need to re-visit the server-side performance.

The client will ask and wait for its turn to interact with a measurement server. This server will eventually send random data with TCP for 10 seconds. After the download test the client will start sending data to the server. During tests both the client and the server store a record every 50 ms how many bytes were transferred during that time.

For analysis we use the average of last 5 second is stored. We have noticed that usually during the first five seconds the mobile technology get stabile. Also with the average RTT of 151.82 ms the TCP slow start is over. By analyzing download graphs we have noticed that after the first 3 seconds the speed usually reach a plateau.

If there is no connection to the Internet, the client application will record this problem and uploads the information later when a connection is available. This way we can map areas without proper data connectivity, too.

C. Position

The client application tries to use all available methods for getting the location. The positioning is possible by GPS, cellular network, WLAN and IP address. The average positioning is very inaccurate, with average accuracy of 1179 meters. However 20% of the measurement have accuracy less than 20 meters and 36% have accuracy less than 100 meters. All of these high accuracy measurement were based on WLAN or GPS positioning.

We are able to store the whole physical route which the user device moved during the measurement. Because the position

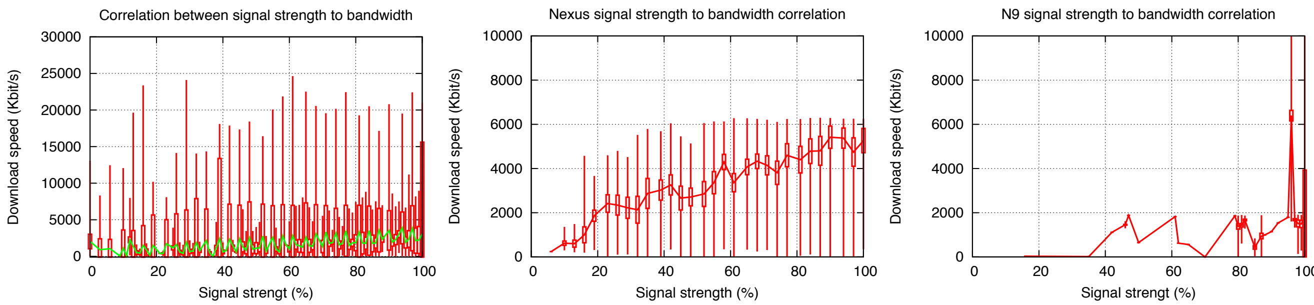

Fig. 1. Signal strength and TCP download speed correlation.

accuracy improves when more satellites become available, we use the last position as reference point for the measurement. The application is also able to get and store movement speed from the positioning system. The website presents only the averaged date per predefined areas to anonymize the results.

D. *Other values*

The application stores base station information, signal strength and radio technology. Every change is stored with millisecond accuracy. This information is stored during the measurement but also passively in the background if the user so chooses; the values are transmitted to the server during the next measurement upload process.

The system is also measuring two-way latency. The client sends ping-like messages with UDP to calculate the RTT before and after the actual data transfer. Both the RTT value and receiving timestamp are stored for every UDP packet. With the receiving timestamp it's possible to do research on packet inter-arrival times. In addition, the server is storing all RTT values from the TCP stack during the transmission. These TCP RTT values allow to analyze congestion impacts on latency.

IV. RESULTS

In this section we analyze different factors that affect the mobile bandwidth. We have identified six possible reasons that limit the receiving bandwidth: radio technology, coverage, congestion caused by other users, mobile phone hardware and software, handovers and seemingly random events.

A. *Different factors*

The first and easiest factor to understand is the radio technology. Newer radio protocols are faster and the maximum data transmission speed is limited directly by the radio technology. With radio technologies, such as HSPA or UMTS, one can estimate the upper limit for possible bandwidth.

Probably the most researched factor is the radio coverage (factor two). The signal strength or lack of it is a natural limiting factor for bandwidth. Distance to base station is one cause for low signal. Another is fading from the buildings and environment. Also structures, especially inside buildings, are blocking the signal.

The third bandwidth limiting factor is congestion caused by other users, due to the provisioning by the operator. If an operator provides 24 Mbit/s link to a base station that usually has 24 users transmitting full speed, then one user is only able to get 1 Mbit/s on average.

The range of different mobile phones is huge (factor four). There are at least five totally different mobile phone platforms and over ten major mobile phone manufacturers using a wide range of platform chipsets. The price range is from thirty dollars to almost one thousand dollars. It's a sensible question to ask, how much the end user device limits the possible download speeds?

User movement might be one source of bandwidth limitation. The probability of handover rises (factor five), causing some impact. Even though the normal movement speed less than 200 km/h should not affect the signal directly in 3G, it's possible that the movement causes a user to hit uncovered areas often and thus get a weak signal.

And last there is still randomness caused by the radio, user context, network hardware and protocols. It's impossible to assert most of these causes. It might be possible to identify more factors from here, but at least we are able to set some limits and deviation of how much these will affect. One might argue that randomness is not a factor, yet, we seek to identify concrete factors from this group in the future studies.

B. *Technology*

The radio technology sets the upper limit for possible bandwidth. The clients are able to identify the following types: GPRS, EDGE, UMTS, HSPA, HSPA+ and LTE. Mobile devices are constantly changing the used technology and the Netradar application is storing this type every time it changes. For this paper we used the most advanced technology available for TCP connections.

There are limitation between phones. We have noticed that, e.g., the Nexus S is reporting UMTS for every 3G radio types. It's visible that the speed is in the range of HSPA or even HSPA+, but the phone is still reporting UMTS. Also some Android phones have reported EDGE when the receiving speed is clearly indicating a good 3G network. Interestingly Meego seems to report the technology with high accuracy, all the reported speed and technologies are within their theoretical limits.

Table I presents maximum and average speeds by technology types. We decided to bundle together UMTS, HSPA and HSPA+ as 3G due to the platforms' reporting limitations. Also 2G consist of both EDGE and GPRS. LTE is listed as it's own.

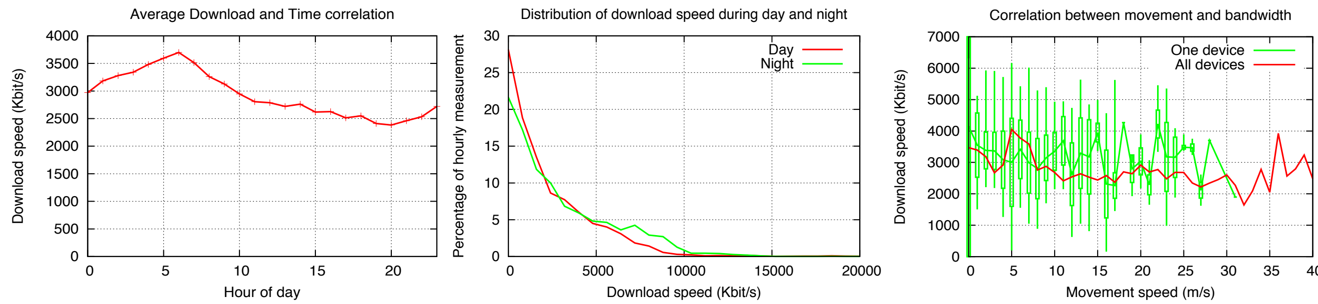

Fig. 2. Time of day correlation to download speed. All data and single 1 Mbit/s limited device in one spot. Movement speed to download speed correlation.

It's visible that every generation allows many times faster transfer speeds compared to the older one. The jump from 2G to 3G is large. LTE is still under development but both the average and max speeds are promising. On the average LTE is about 10 faster than the 3G and the max speed is more than three times.

TABLE I. DOWNLOAD SPEEDS WITH DIFFERENT TECHNOLOGIES.

Technology	Max	Average
2G	213.6 Kbit/s	70.4 Kbit/s
3G	28369.6 Kbit/s	2908.8 Kbit/s
LTE	93471.2 Kbit/s	28818.4 Kbit/s

It's clear that the mobile radio technology used is the most significant factor for the bandwidth level. Thus, the available mobile radio technology is the key aspect when considering e.g. coverage maps and Quality of Experience. Yet, as discussed later, a good radio technology and signal quality does not directly imply great bandwidth.

C. *Signal Strength*

Coverage is the second most important aspect when analyzing the factors behind received bandwidth. We use signal strength to map the coverage. It's possible to gather signal strength from the mobile phone APIs. However, the signal strength we get is heavily averaged and a manufacturer dependent view. We have analyzed the correlation between signal strength and TCP bandwidth earlier in [9]. Usually the measured signal strength has poor correlation with bandwidth. This is due to many reasons, e.g., signal strength is calculated very differently in various mobile phones and it is averaged heavily thus hiding sporadic changes within seconds.

Figure 1 has three graphs mapping signal strength to TCP download speed. The green line represents the average and the red line the minimum and maximum values. The red box around average is the deviation. To avoid the heavy averaging made by the mobile phones we only took samples where there was no movement. This way we can ensure that the signal strength stays about the same during the measurement. For this analysis the last signal strength value during the download was used. First of the graphs has all data from our database. There is correlation with the signal strength, but as stated in earlier research, the deviation is huge.

The second graph has signal strength mapped to TCP download speed from one mobile phone, Samsung Galaxy Nexus running Android. The data is from a single user and one device without operator speed limitation. We have filtered measurements where the user movement speed is less than 1 m/s. The device is reporting whether the network technology is UMTS or EDGE. We wanted to have 3G measurements, thus picking only UMTS. It had 1273 measurements with this filtering. We chose to show this device because the reported signal strength has the best correlation with the download speed.

The third graph has the same filtering for another phone with 390 measurements. We can see that the signal strength has very poor to no correlation with TCP download speed. This presents the problem with signal strength mapping with mobile phones: different platforms and manufacturers calculate and present the data very inconsistently. Thus, it seems that signal strength is a possible metric in some platform and less so in some others. This problem has been reported with some PCMCIA modems too [11] and older smartphones [4]. We recommend that future studies should use Android Nexus - family for this type of studies to limit the problems mentioned.

When analyzing the Nexus correlation, we see that with 20% signal strength the user is able to get an average of 1600 Kbit/s. With 80% signal strength the device is able get average speed almost to the maximum speed level of 6000 Kbit/s. The averages for all devices are around 800 Kbit/s and 2400 Kbit/s, respectively. Thus it seems that when the mobile radio technology remains the same, the user is able to get 3-4 times increase in bandwidth between bad coverage and good coverage.

D. *Congestion*

The bandwidth variation due to the time of day shows the possible congestion in the network. Most user activity is during the day time, from 9 AM to 8 PM. If the congestion or interference causes problems this should be visible in the correlation graphs. Also it should show how much the overall quality suffers from the congestion and interference caused by other users.

Figure 2 shows us two graphs about how the time of day affects the download speed. The first graph shows the overall variation over time. It is visible that during rush hours, from 9 AM to 6 PM o'clock, the download speed is around 1200 Kbit/s lower than during the nightly hours. This is in line with [6].

To understand the difference between congested and uncongested times we chose two different hours. The 3-4 AM represents uncongested nightly hour and 3-4 PM represents daily congested hour. The graph 2 in Figure 2 shows the distribution of download speeds during these hours. During night it is more probable to get over 4000 Kbit/s download speeds.

To further analyse how much the congestion affects the users we chose three users with different speed bottlenecks. One low-end device, one with a maximum data speed limitation and one new device without any limitations. Figure 3 shows these three graphs.

First graph shows a cheaper and older device that gets on average about 3200 Kbit/s download speed during night. The second graph shows a device that has static 1000 Kbit/s (1 Mbit/s) limitation set by the operator. We chose one static location for both devices to limit the affecting factors.

It is visible in the graphs that there is no statistical distinction between day and night. This would indicate that congestion or interference does not affect these devices during the day. The bottleneck is clearly above the operator limit for the second device. For the first device there are small, around 400 Kbit/s, differences between hours, but those are well within standard deviation. It would seem that this area has enough provisioned capacity for 3200 Kbit/s connection.

The third graph shows an unlimited subscription plan on a fast device. It is clear that during the night the user is able to get almost double the speed compared to day time. Interestingly enough the daily average is about the same as with the low end device and within 800 Kbit/s of the overall average. This indicates that the speed is limited to around 3200 Kbit/s by the network during the day. The cause might be interference caused by other users and congestion in the network because of under provisioning.

E. Device

Table II shows the maximum possible download speed reachable by different devices. The first column shows the device name given by the phone itself. Second column is the best single maximum speed recorded by such device. Third and fourth columns have average of 10 and 100 best values. The devices were chosen such that at least 1000 measurements were available. The table is ordered by maximum speed measured with the device.

The table has separated the 3G only and LTE capable devices. The maximum receiving speed with LTE supported devices is very high compared to 3G only devices. The transfer is done with one TCP connection. The speed of the device is limited by the operating system, network stack implementation, hardware chipset and antenna design.

There are couple of surprising devices, such as the Nokia N900 with the Maemo operating system. It's the oldest system, with an abandoned operating system. On the other hand, the downlink speed is above some quite new smart phones.

It's surprising that there is a five times difference between the worst and best device when comparing maximum download speeds, in particular because there are some newer more expensive devices with a low speed such as the Acro S and older devices like N9 on the fast end. The fastest and slowest device are both running Android. The processor speed, nor the operating system are able explain this. The number of measurements is high enough for the comparison to be reliable and also similar devices like N9 and N950 are getting about the same maximum speeds.

TABLE II. MAXIMUM DOWNLOAD SPEEDS (KBIT/S) PER DEVICE.

3G			
Name	Max	Avg. 10 best	Avg. 100 best
Samsung Galaxy Gio	6097.44	6031.09	5680.33
ZTE Blade	6287.20	6021.24	5355.86
Google Nexus S	6315.68	6309.57	6280.83
Motorola Defy+	6321.92	6308.51	6274.27
Samsung Galaxy Tab 7.0	6337.20	6259.75	5900.86
Nokia C7	7518.32	6930.06	6113.24
Sony Xperia acro S	7660.72	6618.54	5181.30
Nokia E7	7730.96	7501.09	7139.63
Nokia N8	7785.28	7334.22	6896.61
Huawei U8800	8646.00	6434.42	5424.11
Motorola Defy	9364.72	6533.69	5776.94
HTC Sensation Z710e	9728.72	9341.91	7956.13
HTC EVO 3D X515m	9972.48	8625.34	7367.04
HTC Vision	10172.24	9907.63	8622.98
Sony Xperia U	10542.96	9953.58	7821.87
Samsung Galaxy Tab 8.9	10946.32	9010.74	6617.68
Samsung Galaxy Xcover	11436.96	7419.80	6334.23
HTC Desire HD	12044.96	10241.74	9921.65
Nokia 808 PureView	12615.68	11904.46	10090.32
Nokia N900	12613.44	9645.96	8236.86
Samsung Galaxy Tab 10.1	12918.88	12031.26	9520.13
Samsung Galaxy S	13839.68	12158.96	7390.49
Samsung Galaxy Nexus	14388.16	11438.23	9105.00
Samsung Galaxy Note	16486.80	10882.33	9357.16
HTC Desire	16517.20	10503.42	6018.41
HTC Desire HD A9191	16739.20	12336.09	10328.96
Samsung Galaxy S 3	17647.84	16396.13	15150.31
Samsung Galaxy S	21971.76	10830.26	8266.92
Nokia N950	22766.72	19036.01	9123.46
Nokia N9	24388.40	21491.48	14487.79
Samsung Galaxy S 2	28369.52	16792.03	11772.77
LTE			
Samsung Galaxy Tab 8.9 LTE	54411.76	45660.20	38955.61
Apple iPhone	60176.00	55486.40	45165.52

F. Handovers

Handovers are typically associated to movement of the end user and his device. Yet, handovers can also happen in stationary positions, i.e., even if the device is not moving, it can do handovers. Handovers in such situations can be triggered by the device when searching for a better base station or by the network, when balancing the load of different base stations. A handover can be between base stations of the same technology (horizontal handovers) or they can trigger a change in the radio technology (vertical handovers), e.g. move from 2G to 3G or 3G to LTE.

When the user is moving, our analysis shows that the movement itself does not affect the quality of the connectivity. Yet, mobility of the device creates handovers, both vertical and horizontal, that result in sporadic breaks in the connectivity and cause delays and potentially packet loss to data transfers.

Horizontal handovers are usually caused by user mobility. To analyze the impact of movement, we can analyze measurements with different movement speeds. It's clearly visible on our map that highways and major roads are very well mapped by our users. Figure 2 graph 3 shows correlation between movement speed and download speed. The movement is recorded from the device during the measurement. All our

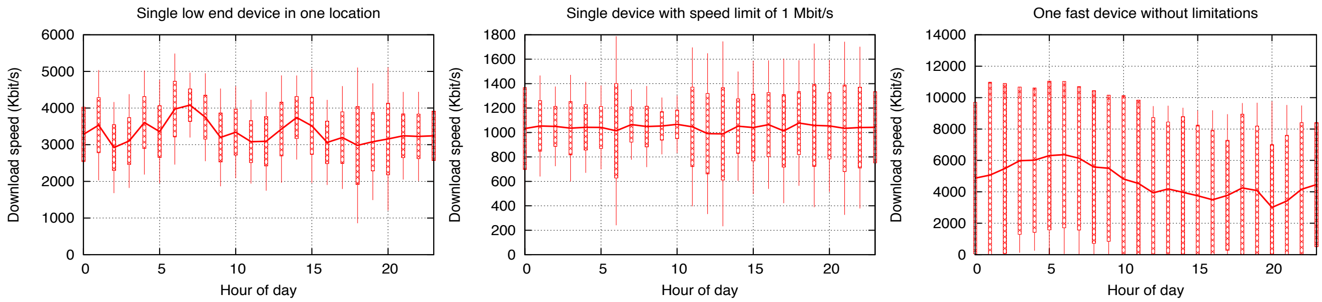

Fig. 3. Time of day correlation to download speed in one location. Device with no max speed limitation and device with max 1 Mbit/s limit.

platforms are able to give one meter per second accuracy for the movement.

The green line shows one device correlation from movement speed to bandwidth. The deviation and min/max are included. The red line shows overall correlation with all the devices and users from our database. The average download speed when user is not moving at all is 3472 Kbit/s. The deviation is huge, from 0 to 52784 Kbit/s. The second graph has one device without a speed limit. There is some fluctuation when the speed is higher, though, the change is within the deviation.

Movement should cause more horizontal handovers: switching from base station to another. The average download speed is 3048 Kbit/s with no handovers and 1864 Kbit/s handovers happen during the connection. Handover is five times more likely if moving 30 km/h compared to staying still. This explains the variation seen in Figure 2 graph 3. These results are in line with movement speed to signal strength correlation done in [9].

When there is a horizontal handover, the impact is clear, the average drops by 1200 Kbit/s. Though only 0.046% of the measurements did have horizontal handover, thus keeping the overall impact small.

We have not analyzed the reason of vertical handovers but only the impact. When there is at least one technology change, the average bandwidth is 1258.4 Kbit/s. With no vertical handovers the average is already 2943.2 Kbit/s. Vertical handovers are much more frequent than horizontal ones. This is due to the way the technology is chosen. The base station forces the mobile client to change the technology to test it. The search is done during connection establishment, our analyzes suggest during the first five seconds. If the connection is not stable the search is repeated. 0.3% of the measurements have reported this kind of vertical handover during calculating the download average. There are deeper studies about the impact of vertical handovers between 2G and 3G in [3] and [1].

V. Conclusion

This paper has presented our system called Netradar, designed to help end users to gather and share data about mobile networks and devices, and to support researchers in understanding better the complex world of mobile network connectivity. We presented as case study the received bandwidth quality and argued that we can explain much of the measurements using five factors: radio technology, coverage, congestion caused by other users, mobile phone itself and handovers. The radio technology and coverage were quite obvious to have the most impact. The congestion and thus operator provisioning was more surprising. During rush hours the average bandwidth is cut by a third. The most surprising factor was the device itself. Our hypothesis would have been that the device has an impact but it's limited. Now it seems that a good device is able to get many times the peak bandwidth compared to simpler devices. It's also worthwhile to mention, that the publication year nor the price gives clear indication of the phone receiving capabilities. Our work is just beginning and we can foresee a huge number of interesting questions, where Netradar can provide the missing piece of data to researchers.

References

[1] L. Bhebhe and A. Arjona. Data outage across 3G & 2G wireless networks. In *IEEE WoWMoM*, 2008.

[2] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin. A first look at traffic on smartphones. In *ACM IMC*, 2010.

[3] A. Gurtov and J. Korhonen. Effect of vertical handovers on performance of TCP-friendly rate control. *SIGMOBILE Mob. Comput. Commun. Rev.*, 8(3):73–87, July 2004.

[4] J. Huang, Q. Xu, B. Tiwana, Z. M. Mao, M. Zhang, and P. Bahl. Anatomizing application performance differences on smartphones. In *ACM Mobisys*, 2010.

[5] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson. Netalyzr: illuminating the edge network. In *ACM IMC*, 2010.

[6] X. Liu, A. Sridharan, S. Machiraju, M. Seshadri, and H. Zang. Experiences in a 3G network: interplay between the wireless channel and applications. In *ACM Mobicom*, 2008.

[7] S. Sen, J. Yoon, J. Hare, J. Ormont, and S. Banerjee. Can they hear me now?: a case for a client-assisted approach to monitoring wide-area wireless networks. In *ACM IMC*, 2011.

[8] V. Singh, J. Ott, and I. Curcio. Predictive buffering for streaming video in 3G networks. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, pages 1 –10, june 2012.

[9] S. Sonntag, L. Schulte, and J. Manner. Mobile network measurements - It's not all about signal strength. In *Wireless Communications and Networking Conference, 2013. WCNC. 2013 IEEE*, april 2013.

[10] Z. Wang, Z. Qian, Q. Xu, Z. Mao, and M. Zhang. An untold story of middleboxes in cellular networks. *SIGCOMM Comput. Commun. Rev.*, 41(4):374–385, Aug. 2011.

[11] J. Yao, S. S. Kanhere, and M. Hassan. An empirical study of bandwidth predictability in mobile computing. In *Proceedings of the third ACM international workshop on Wireless network testbeds, experimental evaluation and characterization*, WiNTECH '08, pages 11–18, New York, NY, USA, 2008. ACM.