This paper is a preprint (IEEE “accepted” status).

Low-Cost Highly Accurate Energy Profiling of IEEE 802.11n Communication Driver

Cornel Isbiceanu, Marius Marcu, Valentin Stangaciu, Sebastian Fuicu
Department of Computer and Information Technology
Politehnica University of Timisoara
Timisoara, Romania
cornel.isbiceanu@yahoo.com, marius.marcu@cs.upt.ro, valentin.stangaciu@cs.upt.ro, sebastian.fuicu@cs.upt.ro

Abstract— **Energy efficiency aims to reduce the amount of consumed energy required to provide services and functionalities, by utilizing commonly accepted methods to reduce the losses of energy or by adopting a more efficient technology or production process. In case of a wireless network, the devices operating on battery try to follow this idea by reducing the energy they consume without affecting the performance required for a certain task. In order to achieve this goal and further improve the efficiency of already low-power wireless cards a highly accurate energy profiling method is needed. In this paper we propose the methodology for profiling 802.11n standard wireless communication card, from the energy and performance perspectives at high sampling (10k) rates and accuracy (1e-5).**

Keywords*— *wireless communication; energy profiling; high accuracy; high sampling rates

I. INTRODUCTION

One of the most important evolution directions of mobile computing is oriented toward mobile wireless services and applications but they are limited by their battery capacity. Battery technologies evolution could not keep the increasing energy requirements of the new mobile applications and devices, hence the energy efficiency and power consumption still remained one of the main optimization problems of mobile computing. For mobile devices an important role in the device's power consumption profile is taken by the wireless communication chipset, along with the processor and display [1]. There are some substantial power consumption differences between GSM communication, TCP/IP communication over WiFI or Bluetooth communication, which show that certain types of wireless communication are not well suited for mobile devices. Recently, low-power technologies, such as Bluetooth Low-Energy have been introduced to support always connected mobile applications. In case of a wireless network, the devices operating on battery try to follow this idea by reducing the energy they consume without affecting the performance required for a certain task. In order to achieve this goal and further improve the efficiency of already low-power wireless cards a highly accurate energy profiling method is needed. In this paper we propose an offline measurement methodology for profiling 802.11n standard wireless communication card, from the energy and performance perspectives at high sampling (10k) rates and accuracy (1e-5).

WLAN technology energy efficiency improvements span very diverse aspects of 802.11 standard operation, from management procedures and wired infrastructure support, to physical and medium access layers [2]. Determining the energy-efficiency of a device can be achieved either by hardware power measurements or by an energy estimation technique. Energy estimation models enable simple and fast energy calculations and overcomes some of the major limitations of physical measurements, such as cost, time-intensive setup, and hardware dependency. However, the accuracy of the estimation techniques is lower compared to physical power measurements, being thus ineffective when analyzing wireless communication events individually [3]. Measuring energy consumption of wireless communication events is not an easy task due to the time synchronization problems and sensor resolution at higher sampling rates. Addressing this topic, the contributions of this paper consist in (1) measurement test bench including physical setup and software benchmarks; (2) accurate timing synchronization of the measurement probes located at several communication stack layers; and (3) synthesis of measurement results for an IEEE 802.11n wireless USB dongle.

The paper is structured as presented: Section II describes related research work on energy consumption and energy efficiency measurement of wireless interfaces, Section III introduces the measurement methodology, Section IV presents the experimental results analysis and the last section concludes the paper.

II. RELATED WORK

During the last years, numerous studies related to energy efficiency on wireless networks have been published. Most of them are focused on the hardware equipment, which typically depend on batteries and, thus, have constraints in terms of power consumption [4]. The most power consuming applications for mobile devices are the ones that comprise data transmission, particularly via the 3G/LTE interface to access the Internet [5]. For this reason, [5] focuses on optimizing the power consumption by large files download on smartphones. Their work was to compare energy consumption and duration of 3G/WIFI asynchronously and scheduled downloads. In order to automatize measurements, they developed a download tool to perform HTTP downloads, while measuring the ground truth power measurements with the Monsoon power monitoring tool [5]. However, the authors performed measurements at minimum 30s download time to distinguish clearly the phases in power consumption and avoid time synchronization problems.

Mobile devices can also act as mobile access points (AP) - that usually runs at full power - which will dramatically reduce the life time of the battery. The work presented in [6] is an energy saving algorithm for a tethering smartphone that plays a role of mobile AP temporarily. It saves energy by introducing the sleep cycle as in power save mode but successfully keeps clients from transmitting while it sleeps. The main purpose of the algorithm is to reduce power consumption by powering off the wireless interface when no traffic is present [6]. Experimental measurement results of the proposed solution have been achieved using the same real-time power monitor device as used in [5]. However, measurements recorded the power consumption of the entire device, interfering with consumption of other components, therefore reducing the accuracy of the test bench.

In order to study the energy consumption and optimize the energy-efficiency of wireless communication, it is vital to have a better understanding of the different scenarios that may be imposed by the nature of the device. To overcome this situation, different energy measurement and estimation techniques have been proposed, as described in [3]. The proposed methodology for generic measurement of USB wireless interfaces enables the measurement of the energy consumption of a single network interface by employing high precision measurement hardware. By using this methodology, it is possible to study multiple network technologies, which make it possible to compare and study the behavior of distinct access technologies under different scenarios and conditions [3]. Using existing development boards to measure energy involves development boards capable of providing the energy consumption of the network interface alone instead of the overall system. Monitoring the device battery terminals implies intercepting the battery terminals of the mobile device in order to measure the power consumption, adding a shunt resistor, having a stable power source with no variation for the Voltage, and also having the capability to isolate the transmission energy. Most smart devices implement built-in sensors and smart battery interfaces which are capable of power consumption profiling [3].

No matter the measurement methodology used, a thorough understanding of the power consumption profiles of real world wireless interfaces and communication is of paramount importance to develop energy-efficient protocols and optimizations on realistic and accurate energy models [2]. The measurement setup is decided based on the timing and accuracy requirements of the optimization task. The authors of [2] provides an in-depth experimental investigation of the per-frame energy consumption components in 802.11 WLAN devices. The main contribution of [2] is the break-down of the energy cost in specific components of the protocol stack, revealing that a substantial fraction of energy is consumed by the processing of packets throughout the protocol stack.

Energy optimization techniques need to be validated by simulations and measurements. Most of the authors validate their results by using simple or complex energy estimation models, but they lack physical measurement support due to time synchronization and sensing accuracy problems. For example [7] describes how performance counters can be used to estimate energy consumption along with performance measurement. The authors of [8] introduce a real-time detailed energy accounting for wireless sensor nodes, providing capabilities for directly observing energy usage of multiple subsystems. The maximum sampling rate supported by the current sensors used by the platform is 25KH.

An important problem to solve when measuring power is correlation and synchronization with the internal events impacting the power consumption. Authors of [9] implemented a testbed for profiling wireless embedded systems using high-accurate timing information that enables logical events to be correlated with power samples. A power meter for wireless sensors nodes is presented in [10]. It can dynamically adjust the resistance of shunt resistor based on the current value and transmit the real-time power measurements host node through the power line, by modulating the current load and the supply voltage.

In our work we address wireless communication at several layers of communication and various protocols from power consumption perspective. We use a digital multimeter to measure current consumption at the physical interface of a USB wireless dongle while sending one packet of information. We tried to track accurately the packet at various observable interfaces: USB, RADIO, LAN infrastructure or TCP/IP stack.

III. Measurement Methodology

The measurement methodology used in this research was chosen in order to measure energy consumption by any kind of USB Wireless Interface. We decided to use external USB network interfaces, because we can measure very accurately the energy consumed only by the interface. Given the fact that in general the energy consumption for a wireless interfaces has slight variations in time it is necessary to use a measurement device that is capable of providing consistency, high sampling rates and high precision measurements. Fig. 1 shows the overall measurement test bench setup, allowing independent synchronized measurements of each component.

The test bench, in terms of hardware, is composed out of (1) a high precision digital multimeter that is capable of measuring very accurately energy consumption of the USB network interface, (2) a wireless packet capture device used to monitor the time the packets are actually received/sent by/to the access point, (3), (4) two computers required to eliminate any USB interference, (5) a wireless access point, (6) an external and stable power supply for the USB network interface and (7) the device under measurement – the USB network interface. The external power supply for the USB network interface was added after the initial sets of measurements, when we observed a significant difference of voltage in the power line of the USB based on the computer model, number of connected USB devices and power consumption of the connected USB devices to the computer.

Given the fact that it is necessary to have repeatable and accurate tests, the following software have been used:

- *Wireshark*: for analyzing the captured packets from USB interface, LAN interface, *AirPCapNx* and *USBPcap*.
- *AirPCapNx*: is a wireless adapter used to capture and analyze IEEE 802.11 wireless radio traffic.
- *USBPcap*: in order to capture the packets sent by the USB device.
- *PacketSender*: for receiving/sending TCP and UDP packets using various test cases.
- Keithley Test Script Processor: for creating test scripts run on the measurement device in order to synchronize its measurements to the ones performed by other equipment in the testing workbench.

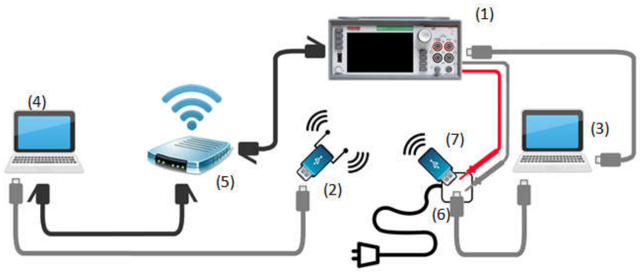

Figure 1. Energy measurement setup

The methodology that we proposed and implemented supports the possibility of studying different network technologies, which makes it possible to study and compare the behavior of distinct access technologies under different conditions and scenarios.

A. *Device synchronization*

Given the fact that a lot of our measurements are performed by several devices, that makes them dependent on the device's internal time, it is essential that they are synchronized. In order to achieve this requirement both the computers and digital multimeter need to be connected to the same network. First, the time between the computers are synchronized and afterwards the digital multimeter. It is notable to mention that the digital multimeter only allows for the time to be synced with a granularity of one second, so some time offset may occur that will need to be corrected during the data processing.

The time for both computers is synchronized using the OS command prompt using Network Time Protocol (NTP) and for the multimeter with the help of the web interface and a custom Keithley's Test Script.

B. *Test methodology*

We started by setting up the access point to use a radio channel that is not in use and is not overlapping over existing radio channels, in order to reduce interference and improve the overall signal of the network.

For the measurements that were conducted the authentication method was set to Open System. Before any devices are connected to the network, it is essential that *AirPCapNx* is configured to use the same channel used by the access point. Afterwards the devices network adapters can be connected to the access point and Wireshark can be open to start capturing the packets.

The IPv4 addresses provided by the access point to the network adapters have been used in order to send packet from one network adapter to the other one. It is necessary that we set an unused port by the system and configure it in the Firewall both for Inbound Rules and Outbound Rules. The time for both computers is synchronized as mentioned.

The Keithley DMM7510 network interface is connected to the access point and a custom test script (TSP) file for Keithley's Test Script Processor is loaded on the device. The TSP file has functions that will allow for the device time to be externally set and also for the measurements to be captured and stored on an external device.

For improving the time offset between the computers and the Keithley DMM7510, the difference between the times received from the multimeter and the computer time is and the duration off the time it takes to receive a response is used as an offset. The USBPcap is configured and started to record all the packets send by the USB network interface.

PacketSender application is used for receiving/sending packets, and allows for specifying the packet content in ASCII/HEX, the IP and port the packet should be send, the number of time the packet should be resend and protocol. After a test has been completed, Wireshark has the timestamp of the packets that have been received/send, *USBPcap* has the timestamp of the packets that (it's necessary to filter out the data and remain with the packets that were sent using *PacketSender*) have been send to the USB Network Interface, Keithley DMM7510 have the measurements for the energy consumption over time, *AirPCap* also has the timestamp of the packets (it's necessary to filter out the data and keep with the packets that were sent using *PacketSender*) have actually been received/send from/to the USB Network Interface.

All the collected information are stored in Comma Separated Value (CSV) files for later processing. In order to process all the measured information, a set of scripts were built to extract the data and stored it for later use in a database. A web application has been developed to analyze the data.

By using all the collected information we can have a better understanding of the energy consumption for a Wireless Network in different scenarios.

C. *Set of tests*

In order to analyze energy consumption of WiFi communication we performed several tests. Each test is a combination of several parameters:

- Transport protocol: UDP or TCP
- Direction: incoming or outgoing packets
- Rate: Number of packets per second
- Distance between sender and receiver
- Packet Size: a packet size is fixed and equal to the maximum transmission unit (MTU).

Next, several tests are mentioned as examples:

- UDP packets send/received from various distances with the same transmission rate and a fixed packet size.

- UDP packets send/received from a fixed distance with different transmission rate and a fixed packet size. The following transmission rates are consider:
  - low : 1 packet/second
  - medium: 10 packets/second
  - high : 100 packets/second

- UDP packets send/received from different access points. The following Access Points were used:
  - Asus RT-AC56U
  - Tenda W316R

- UDP packets send/received with different signal strength, same transmission rate and a fixed packet size. The following signal strengths levels have been consider and used during tests executions:
  - low: 10%
  - medium: 60%
  - high: 100 %

- UDP/TCP packets send/received from a fixed distance, with different transmission rate, fixed packet size and constant signal strength. The following transmission rates are consider:
    - low: 1 packet/second
    - medium: 10 packets/second
    - high: 100 packets/second

TCP and UDP packets.

D. Data analysis

In order to process the big amount of data resulted during a measurement test and, later on, to correlate all the events that are logged, we propose the following web application architecture, presented in Fig. 2. Because of the time and resource limitation of processing data on a single computer, we decided to design the prototype as a web application. This implies connectivity to Internet network, which allows also sharing the results with multiple users.

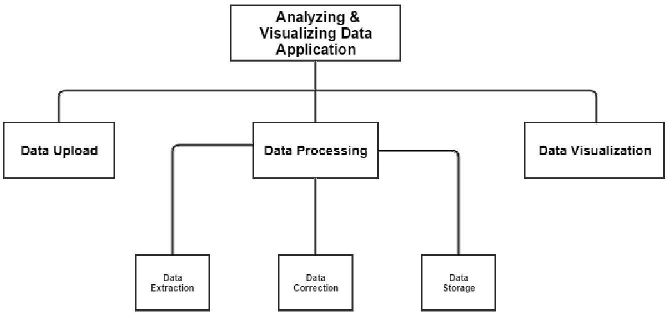

Figure 2 TCP packet reception power consumption

The Data Upload module function is to upload the files generated by testing equipment to the server in an intuitive and easy mode. It does this by offering the user the option to drag and drop the files to a specify area. In order to identify that the user has uploaded all the necessary files, a check function is called that will check for the files extension to be CSV and will check for specific keywords in the files name e.g.: multimeter.csv, lan123.csv. Once the files are validated the Data Upload module checks the presents of keys in the files name (e.g.: lan123.csv the key is 123). The key is a concept used to create a directory on the server where all the unprocessed data is stored. In case the key is not present a random 16 character key is generated.

The Data Processing module is composed out of three sub-modules: Data Extraction, Data Correction and Data Storage. The first module has the role to extract the data from the .csv files and to associate them with the corresponding events. This is done by applying a specific parsing function depending on the file name. In case one of the files is not present or has corrupted data, the sub-module will cancel the parsing of the measurement test and, later, when the user tries to access the Data Visualization, it will receive a notification with the issue. In case the data processing module manages successfully to extract the data, it will send the data to the next sub-module. In Data Correction sub-module all the data provided by the multimeter is correlated with events coming from USB, WiFi, LAN or AirPCap. In case there is any missing or erroneous value for a specific moment in time corresponding to an event, the module will construct the missing data using linear interpolation. The last sub-module has the function of storing the data in a relational database for further use. It does this by first creating tables for the corresponding events. The table name is formed from the event name and the key provided by the Data Upload module. Any existing table will be replaced by the corresponding new one. For optimization purposes specific indexes are created for corresponding tables.

The Data Visualization module has the goal of providing the user a current consumption chart with the corresponding events mapped to the specific moment in time (Fig. 3). The user can visualize specific measurement tests by accessing a URL with the measurement key embedded in it. For analyzing only a specific part of the measurement test the user will select the interest area and later the chart will be drawn with the corresponding data. There are also options to download the chart in one of the following formats: PNG, JPEG, PDF and SVG.

An event is associated to the moment when the packet is send/received to/by the components of the test bench. The four probe points we measure and transform in visual events are identified by (Fig. 3): L – LAN; W – WiFi; U – USB; A – AirPCap.

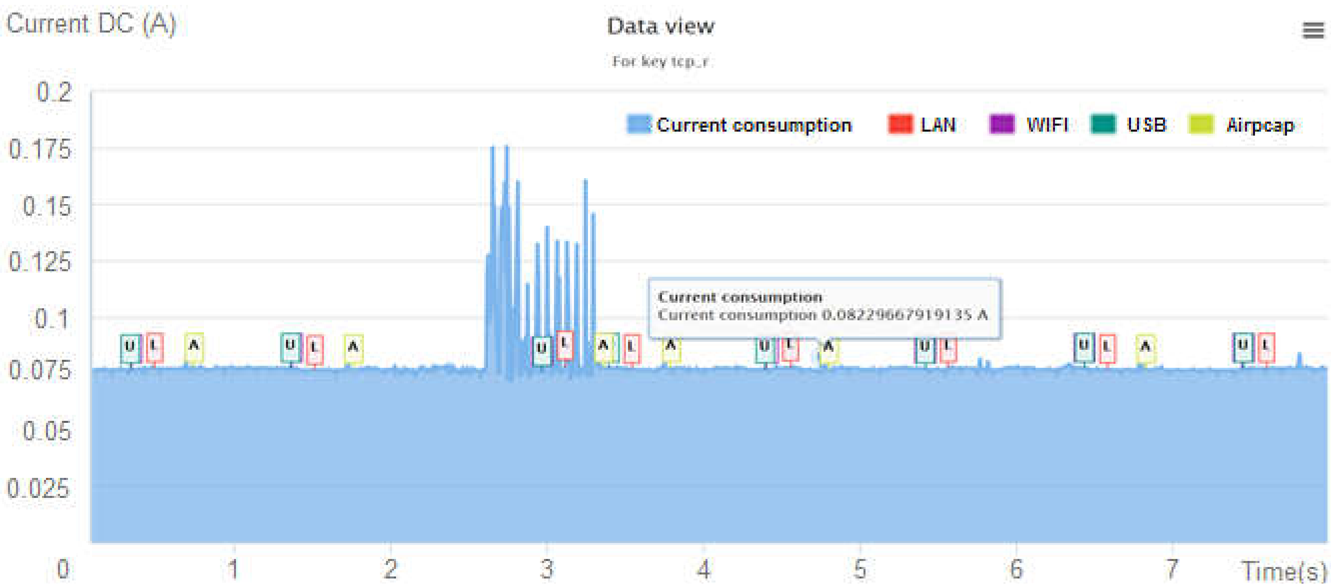

Figure 3 Energy and events analysis tool

IV. Experimental Results

Four types of events are mapped on the power consumption chart in order to identify WiFi communication activity. Thus we can identify very precisely the time and energy consumed by every packet.

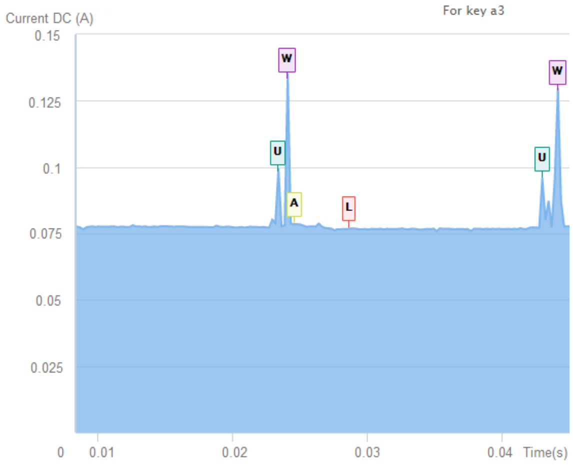

Figure 4. Current for transmission at a rate of 50 packets/sec

The four events shown in Fig. 3 are better seen when we zoom in the chart (Fig. 4 and 5): USB packet send/receive (U), WiFi reception/transmission (W), Airpcap sniffing (A) and LAN reception/transmission (L). During test execution, synchronization between the events is required. Short misalignments between events and measurement are acceptable because of the very short time intervals (see Fig. 4 and Fig. 5). However, we can identify the power consumption spikes, being in correlation with USB packet send/receive and LAN reception/transmission events. From Fig. 4 and Fig. 5 we can see that the events order is repeatable and constant. However is memorable to mention that depending on the transfer rate and the packets size some packets were not capture by AirPCapNx.

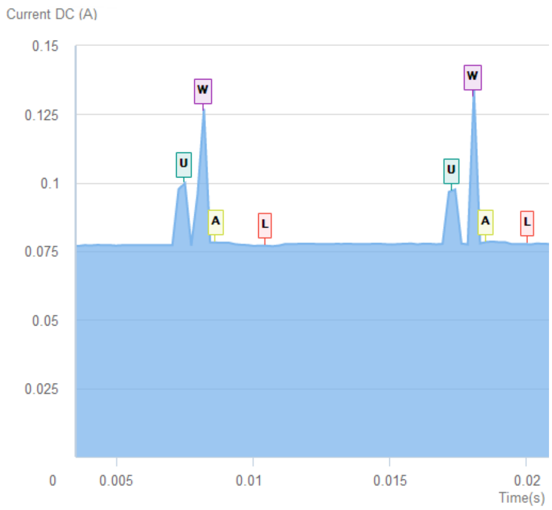

Figure 5. Current for transmission at a rate of 100 packets/sec

During the time of developing the test bench we did a total of 72 tests, but only 33 offered relevant data. These tests are a combination of transmission rate, distance, packet size, USB network interface, direction, signal strength. During all tests the packet size was set to MTU. Below we analyze the power consumption difference when sending data from a fix position with excellent signal quality, but with different transmission rate.

	1 packet / sec	50 packets / sec	100 packets / sec
Idle	0.07646 A	0.07660 A	0.07662 A
Transmission	0.07763 A	0.12621 A	0.13165 A
Reception	0.07762 A	0.11787 A	0.11976 A
Max Spike	0.08998 A	0.14939 A	0.14965 A

Table I Average Current Consumption during different transmission rates

In Table I is presented the current consumption levels and we can observe a significant difference between the transmission rate of 1 packet/sec and 50 packets/sec. Thus, we observe that as the quantity of packets increases in the transmission rate, so does the current consumption. We can also see a difference in current consumption during the transmission and reception stage, as well as the max current consumption spike found during each transmission rate.

Next, we propose to test if there is any significant difference when running the previous test set, with different signal strengths. In total were run 30 tests at transmission rate of 50 packets/sec during different signal strengths. We removed all the 4 events mapped on the current chart for a better perspective view.

From Table II we can see that there are differences in the current consumption during each test. The most memorable ones being when the signal strength is lower than 4 out of 5 lines and a transmission rate higher or equal to 50 packets/seconds are used. The measurements are represented in Fig. 6.

Signal Strength	Rate (packets/sec)	Reception(A)	Transmission(A)
1/5	1	0.0775695	0.0776529
2/5	1	0.0775156	0.0777055
3/5	1	0.0774850	0.0776312
4/5	1	0.0774318	0.0776203
5/5	1	0.0774317	0.0774122
1/5	50	0.1857067	0.1951695
2/5	50	0.1641983	0.1889783
3/5	50	0.1636079	0.1777105
4/5	50	0.1076408	0.1341243
5/5	50	0.1032889	0.1239021
1/5	100	0.1956710	0.2043947
2/5	100	0.1908963	0.2053898
3/5	100	0.1868961	0.2050446
4/5	100	0.1096292	0.1371763
5/5	100	0.1043158	0.1306565

Table II Average current consumption during different signal strengths

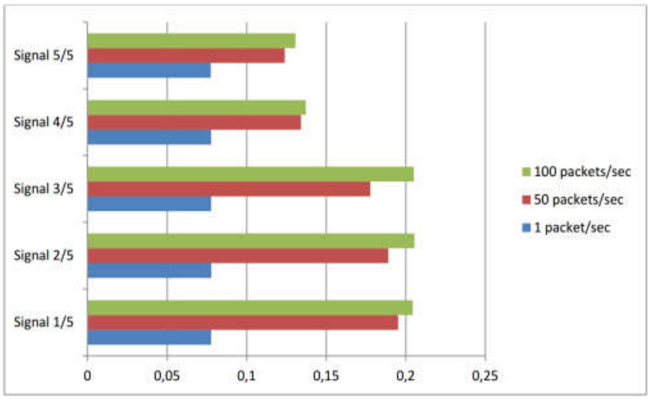

Figure 6. Average current consumption during transmission with different signal strengths [A]

When sending packets at very low rates (1 packet/sec) we cannot put in evidence any difference in power consumption function of distance between sender and receiver. However, increasing the transmission rate we can observe that transmission current is almost 50% higher for low signal strengths (1-3/5) compared with high signal strengths (4-5/5) as shown in Fig. 6.

Next, we wanted to see the difference in power consumption if we use a different USB network interface and run similar tests under the same scenarios. Table III shows the difference, which are significantly higher.

USB Network Interface	Rate (packets/sec)	Transmission Speed	Transmission(A)
DWA-121 150 Mbps	1	14 Mbps	0.0777055
	50		0.1889783
	100		0.2053898
TL-WN821N 300 Mbps	1	150 Mbps	0.2244123
	50		0.4483153
	100		0.4518918

Table III Current consumption difference between two distinct USB Network Interfaces

Lastly, we wanted to see the difference in power consumption if we change the channel width from 20 MHz to 40 MHz over 2.4 GHz frequency band, with excellent signal quality (Table IV). Fig. 7 shows the difference, which are higher at higher transmission packet rates. When using 40Hz channel width aproximatively 35% increase in power consumption has been measured at 100 packets/sec transmission rate.

Channel Width	Rate (packets/sec)	Transmission(A)
20 MHz	15	0.0789273
	25	0.0793910
	50	0.0799777
	100	0.0811061
40 MHz	15	0.0803748
	25	0.0807898
	50	0.0943531
	100	0.1091167

Table IV. Current consumption difference between channel widths

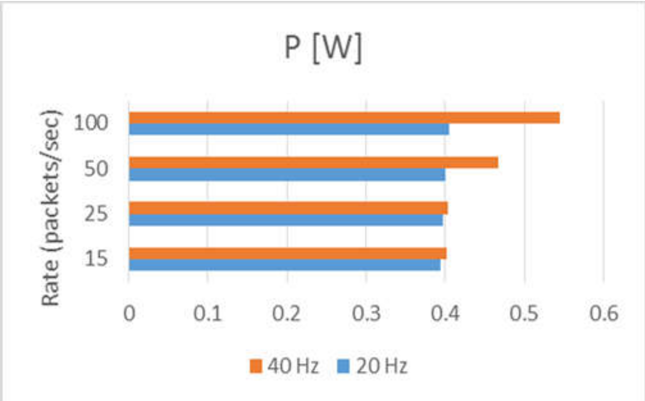

Figure 7. Average power consumption during transmission at different packet rates and channel widths

V. Conclusions

As mentioned above, a lot of studies were made related to energy efficiency in wireless networks on different aspects. The main focus and goals of the dissertation is to study and improve some of the existing solutions for energy efficiency that focus on the end user. There are different approaches that try to solve the problem. Some of them are: using an energy saving decision algorithm or by setting up a scheduling downloading algorithm. In this context we proposed a highly accurate methodology and tools for power profiling of wireless communication at the packet level. Per packet energy consumption can be computed using the proposed methodology.

A total of 72 tests were made, but only 33 offered relevant data. These tests are a combination of transmission rate, distance, packet size, USB network interface, direction, signal strength and USB Network Interfaces. Some power consumption patterns were observed, but we consider that more tests need to be conducted before a proper conclusion can be made in form of a mathematical model.

References

[1] Narseo Vallina-Rodriguez and Jon Crowcroft, "Energy Management Techniques in Modern Mobile Handsets", IEEE COMMUNICATIONS SURVEYS & TUTORIALS, Vol. 15(1), 2013.

[2] Andres Garcia-Saavedra, Pablo Serrano, Albert Banchs, Giuseppe Bianchi, "Energy Consumption Anatomy of 802.11 Devices and its Implication on Modeling and Design", Co-NEXT'12, Dec. 2012, Nice, France.

[3] Jean-Marc Pierson, "Large-scale Distributed Systems and Energy Efficiency: A Holistic View", John Wiley & Sons, Mar 5, 2015.

[4] Karina Gomez, Cigdem Sengul, Nico Bayer, Roberto Riggio, Tinku Rasheed, Daniele Miorandi, "MORFEO: Saving energy in wireless access infrastructures", World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a, Madrid, pp. 1-6, 4-7 June, 2013.

[5] Selim Ickin, Thomas Zinner, Katarzyna Wac, Markus Fiedler, "Catching the download train: Energy-efficient file downloading on smartphones", 26th International Teletraffic Congress (ITC), Karlskrona, pp. 1-5, Sep. 2014.

[6] Kyoung-Hak Jung, Yuepeng Qi, Chansu Yu, Young-Joo Suh, "Energy efficient Wifi tethering on a smartphone", INFOCOM, 2014 Proceedings IEEE, Toronto, pp. 1357-1365, Apr. 2014.

[7] Vincent M. Weaver, Matt Johnson, Kiran Kasichayanula, James Ralph, Piotr Luszczek, Dan Terpstra, Shirley Moore, "Measuring Energy and Power with PAPI", 41st International Conference on Parallel Processing Workshops (ICPPW), 2012.

[8] Thanos Stathopoulos, Dustin McIntire, William J. Kaiser, "The Energy Endoscope: Real-Time Detailed Energy Accounting for Wireless Sensor Nodes", Proceedings of the 7th international conference on Information processing in sensor networks, IPSN 2008, Apr. 2008.

[9] Roman Lim, Federico Ferrari, Marco Zimmerling, Christoph Walser, Philipp Sommer, Jan Beutel, "FlockLab: a testbed for distributed, synchronized tracing and profiling of wireless embedded systems", Proceedings of the 12th International Conference on Information Processing in Sensor Networks (IPSN), Apr. 2013.

[10] Ruogu Zhou, Guoliang Xing, "Nemo: A High-fidelity Noninvasive Power Meter System for Wireless Sensor Networks", Proceedings of the 12th International Conference on Information Processing in Sensor Networks (IPSN), Apr. 2013.