Modeling Energy Efficiency in Wireless Internet Communication

Enrico Rantala
Nokia Research Center
Helsinki, Finland
enrico-henrik.rantala
@nokia.com

Arto Karppanen
Nokia Research Center
Espoo, Finland
arto.karppanen
@nokia.com

Seppo Granlund
Nokia Research Center
Espoo, Finland
seppo.granlund
@nokia.com

Pasi Sarolahti
Nokia Research Center
Espoo, Finland
pasi.sarolahti
@nokia.com

ABSTRACT

For wireless mobile Internet users the length of the battery life is one of the most important performance factors. The energy efficiency of the data transmission over radio is a key component affecting the battery lifetime. This paper investigates WLAN energy consumption in network communication on a Mobile handset. We introduce an energy model that allows analysis and simulation of the energy efficiency of the Internet protocols on a Wireless Network Interface, and have extended the NS-2 simulation platform to allow investigating the energy consumption of the Radio Modem and the Power Amplifier in WLAN 802.11g network interface of a mobile device. We have also validated our model against measurements on real wireless hardware, and show that the simulation results closely match the real world behavior. We claim to present more detailed and accurate model of the WLAN energy consumption than what is done by the past work that allows designing and optimizing future Internet protocols towards more energy efficient behavior.

Categories and Subject Descriptors

C.4 [**PERFORMANCE OF SYSTEMS**]: Measurement techniques, Modeling techniques, and Performance attributes

General Terms

Measurement, Performance, Experimentation

Keywords

Wireless networking, energy efficiency, TCP, network simulation

1. INTRODUCTION

The length of the battery life in wireless handheld communication device is one of the most important performance criteria for a mobile user. In addition to developing new battery technologies with better energy capacity, the length of the battery life can be extended by reducing the energy consumption in a mobile device. One of the most significant activities that consume energy is transmitting data over a wireless radio interface. The energy efficiency of wireless data transfer can be improved by designing and optimizing the protocols to be more energy-conservative in their use of the wireless interface. As the Internet is to increasing extent accessed by wireless battery powered devices, we think it is important to consider the energy efficiency as an inherent part in protocol design and optimization.

The target of our study is to create an energy consumption model that can be used for Internet protocol energy efficiency simulations and analysis. The two key energy consuming components related to wireless transmission are the host CPU that executes the TCP/IP protocols and applications, and the Wireless Network Interface (WNI) that takes care of the transmission of the packets to the physical medium. This paper focuses on analyzing the WNI energy consumption, and the analysis of CPU energy efficiency is left for future work. To be able to conduct detailed analysis of the protocol energy efficiency, it is important to be able to separately simulate the battery current drawn by the Power Amplifier (PA) and the Radio Modem (ASIC). In addition, from the model validation point of view, it is important to calibrate the model with real measurements for realistic simulation results. To achieve this we have set up a measurement system capable of microsecond level time resolution battery power sampling. We are using the IEEE 802.11g WLAN modem as the radio hardware. Finally, we validated our model by implementing the calibrated model in NS-2 simulator [7]. The simulated energy values gave good match to measured ones.

The energy consumption of mobile handsets has been under extensive research, but there are not many published studies concerning detailed Wireless Network Interface energy consumption and simulation. In earlier related studies [2, 3, 5, 6, 8, 9, 12] the measurement platforms have been based on laptops or PDA devices with WLAN cards rather than mobile handsets. Because we are interested in analysing the detailed component-level energy consumption in an integrated mobile handset, we cannot base our work on the above mentioned articles. However, there are a few work items that have taken a similar approach to ours. According to our knowledge Stemm and Katz [12] did the first study analysing Wireless Network Interface energy consumption, and they presented methods that have been used in many later studies. The work by Ebert et al. [2] also has close relationship to our work, since they discuss Wireless Network Interface measurement, simulation model parameterization and validation. The energy consumption of three TCP versions - Reno, Newreno and SACK has been compared based on measurement experiment and by emulating various network conditions, see Singh [11]. In this paper we provide a perspective on how to evaluate Internet protocols by using our energy model that we intend to extend in our future work.

The remainder of this paper is organized as follows: in Section 2 we introduce the energy model. The measurement system is presented in Section 3. In Section 4 we validate the energy model with NS-2 and finally Section 5 gives the concluding remarks.

2. ENERGY MODEL

2.1 Wireless Network Interface

Our target is to model mobile handset energy consumption drawn by IEEE 802.11g Wireless Network Interface (WNI) i.e. WLAN (Wireless Local Area Network) modem. Concerning the energy consumption, there are two fundamental components in the WNI: the WLAN ASIC (Application-Specific Integrated Circuit) and the PA (Power Amplifier) located in the same chip. Figure 1 illustrates the WNI structure and the components that are relevant to our energy model.

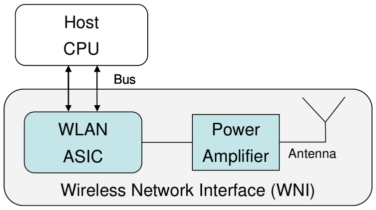

Figure 1. Schematics of the Wireless Network Interface.

The main role of the WLAN ASIC is to run transceiver algorithms and MAC (Medium Access Control) protocol functions. The WLAN ASIC is physically connected to the PA that is part of RF Front-End Module (FEM). Its main task is to amplify transmission signal so that it can be driven to the antenna. Also the host CPU (Central Processing Unit) is shown in Figure 1. The main task of the host CPU is to run applications and protocols. Modeling the energy consumption of the host CPU is left for future work.

In the following we introduce an energy model that encloses separately both fundamental energy consuming WNI components, the WLAN ASIC and the PA. In the following theory we shall use well-known SI-units like Joule for energy, Watt for power, second for time, Volt for voltage and Ampere for current unless otherwise mentioned. The energy consumption of the whole WLAN modem, E_{WNI} , is the sum:

$$E_{WNI} = E_{ASIC} + E_{PA}, \qquad (1)$$

where E_{ASIC} is the energy consumed by the WLAN ASIC (see Section 2.2) and E_{PA} is the energy consumed by the PA (see Section 2.3).

2.2 WLAN ASIC Energy Model

In an active bulk data transfer the WLAN ASIC battery power is fairly constant throughout the experiment (no dramatic difference in receiving, transmitting or idle period). This can be seen in the measurement sample in Figure 2 where transmission periods dominate the file upload. Hence, we shall formulate the WLAN ASIC energy consumption, E_{ASIC}, simply as follows:

$$E_{ASIC} = \overline{P_{ASIC}} \cdot t_{sim}, \qquad (2)$$

where $\overline{P_{ASIC}}$ is the average WLAN ASIC battery power over the simulation time of t_{sim}. For Formula 2 we need to assume that during the experiment, the WLAN ASIC is either in transmission or receiving mode for most of the time, and especially does not enter the sleep mode. In the sleep mode the drawn power drops dramatically for long periods of time. To take into account impact of sleep, idle, receiving and transmitting modes in a simulation, detailed tracing capability is required, which is left for future work.

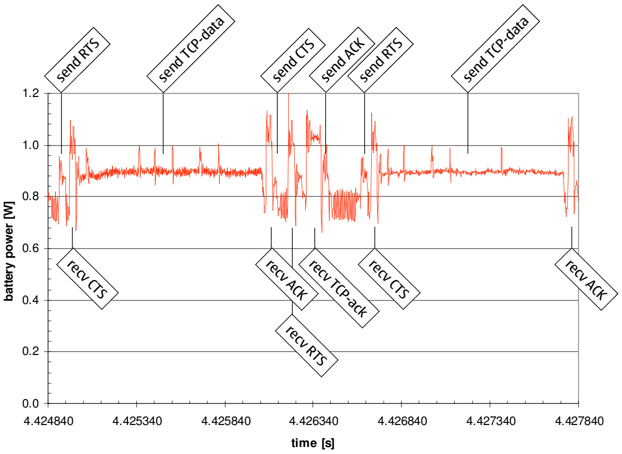

Figure 2. WLAN ASIC battery power measurement sample representing 1500 samples in 3ms time (2   sampling step). The labels point the sent and received WLAN MAC and TCP packets.

2.3 Power Amplifier Energy Model

The PA is highly active only on transmitting periods. On idle periods, only leakage battery current is drawn. The PA battery power measurement sample shown in Figure 3 confirms this.

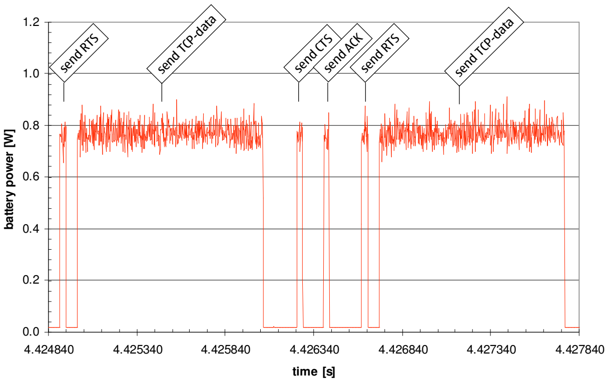

Figure 3. Power Amplifier battery power measurement sample analogous to Figure 2.

We model the PA energy consumption per-packet basis and take also into account idle periods. The energy consumed by the PA, E_{PA}, when n packets are transmitted is as follows:

$$E_{PA} = \sum_{i}^{n} \overline{P_{Tx(i)}^{j}} \cdot t_{TX(i)} + \overline{P_{IDLE}} \cdot \left(t_{sim} - \sum_{i}^{n} t_{TX(i)} \right), \quad (3)$$

where $\overline{P_{TX}}$ is an average packet-specific (selected by *j*) PA battery power (the battery power needed in transmission depends mostly on two packet-specific parameters: PHY (Physical Layer) bit rate and packet size, their impact is considered in [2]), $\overline{P_{IDLE}}$ is the average PA leakage battery power, t_{sim} is the simulation time and t_{TX} is the PA activity time for processing a packet. In order to model t_{TX} we used the analytical formula for 802.11g PHY packet transmission time presented in [1]. However, we had to modify the formula by adding a constant Δ in order to include the impact of hardware implementation.

$$t_{TX}[\ s] = 26\ s + \left\lceil \frac{22 + L[bit]}{4 \cdot R[Mbps]} \right\rceil \cdot 4\ s + \Delta\,, \quad (4)$$

where $\lceil\ \rceil$ is the ceiling function, *L* is the packet length containing all but PLCP (Physical Layer Convergence Protocol), tail, service and pad bits and R is the PHY bit rate in Mbps (see 802.11g ERP OFDM-PHY frame structure and bit rates in [4]). The constant Δ is estimated from the measurements by comparing measured PA activity time to analytical time presented in Formula 4 when Δ = 0. Analysis of various PA activity samples showed that there is a systematic −5 s difference to the analytical time. Thus we concluded Δ = −5 s. The difference originates from 802.11g signal extension time of 6 s (when the PA is not transmitting) and other hardware originated differences to the analytical model like ramp-up and ramp-down times. The mathematics and data analysis behind Δ is left out from this paper. However, a larger study of this paper covering all analysis shall be available in the internet later on.

3. MEASUREMENT AND ANALYSIS

3.1 Measurement Software and Hardware

In order to test the energy model presented in Section 2, we set up a simple reference scenario that can be simulated and measured. The measurement hardware is shown in Figure 4.

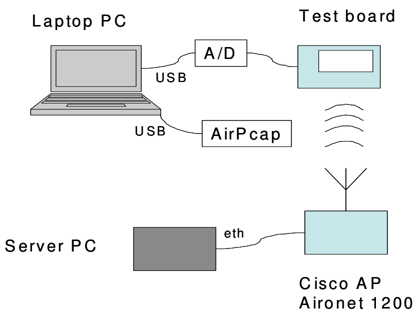

Figure 4. The measurement hardware.

The heart of the hardware is the test board with WLAN modem circuitry and the host CPU running Linux OS. A web browser application runs on top of Linux and the web browser is used for file transmission. A file is stored in the test board's mass memory. This file is large enough to take a couple of seconds of transmission time so that the measurement generates enough trace for statistics. The traffic sent over WLAN can be traced and collected to a trace file. This is done with AirPcap Classic WLAN (802.11b/g) packet capture adapter ("WLAN packet sniffer") and Wireshark protocol analyzer v.1.0.0. The rest of the test system is comprised of Cisco Aironet 1200 Series WLAN AP (Access Point) and a desktop server PC (Windows XP). Server PC contains Xwiki software that can be used for uploading or downloading of files. The WLAN on the test board and Cisco AP are configurable to use fixed PHY bit rate for all communication including Data, RTS, CTS, ACK and Broadcast frames. In addition, RTS threshold can be set manually. The test board contains measurement points that can be used for measuring the battery current drawn by the individual components on the board. The connection method to the test points is shown in Figure 5 that shows how battery current was measured. A 75 milliohm shunt resistor (0R075) is attached to the WLAN ASIC and the PA power supply rails. The A/D converter input was connected over the resistor and the voltage drop over the resistor was measured. This voltage drop is proportional to the current through the circuit.

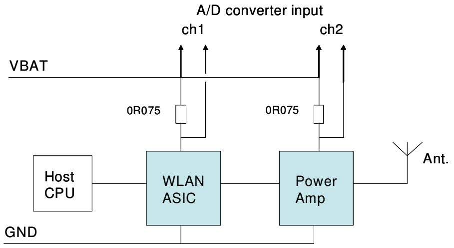

Figure 5. Test board measurement points.

The measurement part consists of the A/D converter that is used for collecting battery current data. We have used National Instruments USB-6251 data acquisition card that has 1 s maximum resolution of 16 bit samples. The measurement data is transferred to the laptop PC via USB cable. The variation of battery voltage can be assumed small so that the instantaneous battery power can be simply obtained from the instantaneous current by the following formula:

$$P_{battery}(t) = U_{battery} \cdot I_{battery}(t)\,. \quad (5)$$

In Formula 5 we have used the voltage value $U_{battery}$ = 3.85 V. This value represents a typical voltage value for a charged lithium-ion mobile handset battery. We do not present a detailed analysis on the accuracy of measured instantaneous current. We only note that the main factors that have impact on the accuracy are the shunt resistor and the A/D converter. Based on manufacturer information, the following accuracy figures are obtained: shunt resistor 1% and A/D converter 0.3%.

3.2 On Measuring

Battery power measurement is favorable to conduct so that it is done simultaneously with AirPcap over-the-air packet capturing. Synchronizing the time stamps of these two traces (possible with reasonable amount of work) helps the analysis when a WLAN packet and the corresponding battery power peak can be matched. The traces are discussed in the following sections.

3.2.1 Measurement Validation and System Parameters

AirPcap trace can be used to validate the measurement in order to see whether the use case is properly conducted and whether the measurement system configuration is as it should be. These can be checked from the AirPcap packet flow and the detailed information on WLAN packets that AirPcap provides. Also by checking the packet IP and MAC addresses, it is possible to see whether there are other WLAN devices around, interfering the measurement. Our file upload measurement case parameters from AirPcap trace are listed in Table 1.

Table 1. Information obtained from AirPcap trace in file upload case. The information is used for measurement validation and simulation parameter harmonization with the measurement system.

Parameter	Value
[a]PHY bit rate	12 Mbps
Mixed 802.11b/g mode	No
WLAN	Pure 802.11g
PHY	[b]ERP-OFDM
RTS/CTS	Always
Delayed ACK	Yes
TCP segment data	1448 bytes
Advertised Window (AP)	64512 bytes
Advertised Window (Mobile)	6912 bytes
MAC segmentation	No
TCP segmentation	No
Power Save Mode (PSM)	No

[a]Fixed PHY bit rate without possibility to link-adaptation, applies to all type of packets (TCP, RTS/CTS, ACK and Broadcast).
[b]Extended Rate PHY - Orthogonal Frequency-Division Multiplexing

3.2.2 Measured Battery Power Estimates

The average WLAN ASIC battery power estimate can be calculated from the WLAN ASIC battery power trace simply by taking average value over the use case. Getting the estimates from the PA battery power trace is not that straightforward. The PA battery power estimates require careful analyzing in order to identify similar packets. Data mining and techniques distinguishing the packets that differ by packet size and PHY bit rate cannot be included inside this paper but shall be available in full-text version of this paper in the internet later on. When the similar packets are found, the battery power estimates can be calculated by averaging over the similar packets. The averaged battery powers are shown in Table 2.

Table 2. Measured average battery powers.

Parameter	Value
ASIC (in file upload)	869 mW
PA (12 Mbps TCP-DATA)	770 mW
PA (12 Mbps RTS)	749 mW
PA (12 Mbps CTS)	746 mW
PA (12 Mbps ACK)	746 mW
PA (when IDLE)	19 mW

4. ENERGY MODEL VALIDATION

In the following we shall validate the energy model by comparing measured and simulated energy consumption figures in an experimental case of single file upload. For energy consumption perspective this is more interesting case than file download because file upload causes more activity on the PA side. For the simulation platform we use NS-2 [7] in which we added our energy model.

4.1 NS-2 and the Simulation Parameters

NS-2 is a discrete, event driven network protocol simulator used widely in research communities. The simulator's OTcl/C++ framework bases on open source model. OTcl (Object-oriented Tool command language) provides a simulation interface to the simulator implementation written in C++. Contribution of the code is done by several universities and research laboratories. NS-2 provides models for various protocols for multiple layers like application, transport, routing, medium (MAC) and physical (PHY) layers. Especially the modeling of the transport layer is creditable; it contains model for UDP (User Datagram Protocol) and models for various TCP (Transmission Control Protocol) flavors like Newreno and Sack. Besides TCP, the support for wireless IEEE 802.11, or in other words, WLAN (Wireless Local Area Network) simulations is important for this work.

Our energy simulation bases on the NS-2 version 2.33. The energy model and related tracing is added to CMU's (Carnegie Mellon University) MAC and PHY implementation. The simulated network infrastructure contains three elements: Wired server, WLAN AP and WLAN mobile, as illustrated in Figure 6.

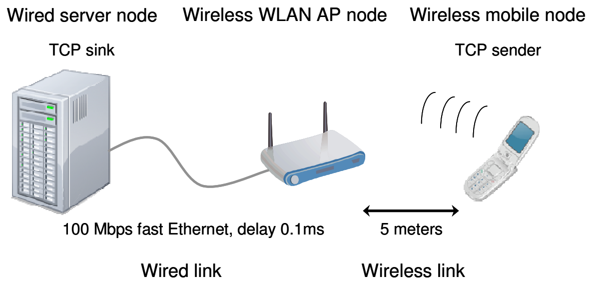

Figure 6. The simulated network infrastructure.

We have taken the WLAN specific parameters for the NS-2 from IEEE 802.11 WLAN specification [4]. These parameters are shown in Table 3 comprising the NS-2 parameterization together with Table 1 and Table 2. Finally, we used default NS-2 parameter values for such parameters which are not shown in the tables.

Table 3. NS-2 high level, TCP, 802.11g ERP-OFDM PHY and channel parameters used for the file upload case.

Parameter	Value
Application	[a]FTP
Medium access mechanism	[b]DCF
Wired/wireless Interface queue	DropTail, priority based
Wireless Interface queue length	50
Routing protocol	[c]DSDV
Propagation	Two-Ray Ground
Antenna	Omni
Mobility	No
[d]Link layer delay	7.5 ms
TCP	One-way NewReno [10], [11]
TCP window size	20 segments
Delayed ACK interval	200 ms
PHY bit rate	12 Mbps
RTSThreshold_	0 bytes
ShortRetryLimit_	7 times
LongRetryLimit_	4 times
CWMin_	15
CWMax_	1023
SlotTime_	9 s
[e]SIFS_	16 s
CPThresh_	10 dB
CSThresh_	1.559e-11 W
RXThresh_	3.652e-10 W
Pt_	0.1 W
freq_	2.4e+9 Hz
maxPropagationDelay_	2 s

[a] File Transfer Protocol

[b] Distributed Coordinate Function.

[c] Destination Sequence Distance Vector.

[d] Link layer delay models hardware related delays caused by e.g. insufficient processing and bus capacity.

[e] Short Interframe Space, actual specification value is 10 s but 6 s of signal extension is added here for practical reasons (to model signal extension in NS-2 without need to modify source code).

4.2 Comparison of Simulated and Measured Results

Figure 7 depicts the flow chart of our simulation and analysis process. The post-processing of an NS-2 trace is straightforward, and any tool appropriate for data parsing and text handling could be used.

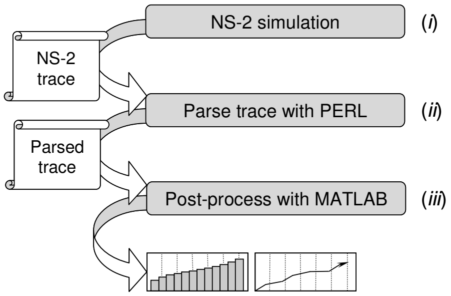

Figure 7. Simulation and analysis flow.

The statistics we selected to compare the simulator and the measurement system operational similarity contains: average goodput and packets per second (to test connection speed), total relative transmission time (to test transmission time calculation) and packet number proportions (to test MAC protocol packet flow). Compared energy related figures are: battery power and energy per bit (both figures are averaged over experiment time). The figures that we compared are collected into Table 4. As can be seen in goodput, packets per second and packet number proportion figures, the operation of the measurement system and the NS-2 simulator is very similar. The energy figures are also very close. We may conclude that the simulator calibration was successful and the model is validated in this case.

Table 4. Compared file upload measurement and simulation.

Quantity	Meas. system	NS-2
Goodput	6.59 Mbps	6.61 Mbps
[a]txtime percentage	63.9 %	63.9 %
Packets/second (all)	1748	1725
Packets/second (TCP)	557	571
TCP packet percentage	33.3 %	33.1 %
RTS packet percentage	33.7 %	33.9 %
CTS packet percentage	16.5 %	16.5 %
ACK packet percentage	16.5 %	16.5 %
Average PA bat. pwr.	498 mW	498 mW
Average PA E per bit	76 nJ/bit	75 nJ/bit
Average ASIC bat. pwr.	869 mW	869 mW
Average ASIC E per bit	132 nJ/bit	132 nJ/bit

[a] Transmission time (or PA activity) proportion of total experiment time.

5. CONCLUDING REMARKS

We have presented an energy model for WLAN 802.11g Wireless Network Interface in a Mobile handset. A new aspect in this approach is that the Power Amplifier and the Radio Modem are modeled as separated entities. Therefore the model presented in this paper is more accurate than the ones presented earlier. Our model was calibrated by a measurement of file upload case of fixed 12 Mbps link of WLAN 802.11g. The calibrated model was validated by comparing simulated energy to the corresponding measurement and a close match was obtained. The comparison of the simulated and measured packet flow statistics also gave a good match. This experiment shows that it is possible to calibrate a simulator by measurements so that the simulated and measured battery energies are very well in line.

As a valuable outcome of our work, we succeeded to build a NS-2 based tool for the future energy efficiency studies. As the next step the authors shall enhance the NS-2 WLAN MAC and PHY layers by adding link adaptation and renewing the radio environment model for more realistic simulation scenarios. This enhanced tool shall be used for the analysis of energy efficiency of the two TCP start-up algorithms: the traditional Slow-Start and challenger approach Quick-Start.

Acknowledgement

This work was supported by TEKES as part of the Future Internet program of TIVIT (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT).

6. REFERENCES

[1] Déziel, M., and Lamont, L. 2005. Implementation of an IEEE 802.11 Link Available Bandwidth Algorithm to Allow Cross-Layering. Wireless And Mobile Computing, Networking And Communications, 2005. (WiMob'2005), IEEE International Conference. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1512893

[2] Ebert, J.P., Aier, S., Kofahl, G., Becker, A., Burns, B., and Wolisz, A. 2002. Measurement and Simulation of the Energy Consumption of an WLAN Interface. TKN Technical Reports Series.

[3] Feeney, L.M., and Nilsson, M. 2001. Investigating the Energy Consumption of a Wireless Network Interface in an Ad hoc Networking Environment. IEEE Infocom 2001, Anchorage, April 2001. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=916651

[4] IEEE Std 802.11™-2007 (Revision of IEEE Std 802.11-1999). Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. http://standards.ieee.org/

[5] Kravets, R., and Krishnan, P. 1998. Power Management Techniques for Mobile Communications. MobiCom'98, Dallas, Texas, October 1998. DOI= http://doi.acm.org/10.1145/288235.288276

[6] Kravets, R., Schwan, K., and Calvert, K. 1999. Power-Aware Communication for Mobile Computers. International Workshop on Mobile Multimedia Communications (MoMuc'99). http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=819474

[7] NS-2 Network Simulator. http://www.isi.edu/nsnam/ns/

[8] Qiao, D., Jain, A., and Shin, K.G. MiSer: An Optimal Low-Energy Transmission Strategy for IEEE 802.11a/h. MobiCom'03, September 14-19, 2003, San Diego, California, USA. DOI= http://doi.acm.org/10.1145/938985.939003

[9] Rizzo, L. 1997. Dummynet: a simple approach to the evaluation of network protocols. ACM Computer Communication Review, Vol.27,n.1,Jan.1997. DOI= http://doi.acm.org/10.1145/251007.251012

[10] Seddik-Ghaleb, A., Ghamri-Doudane, Y., and Senouci, S-M. 2006. A Performance Study of TCP Variants in Terms of Energy Consumption and Average Goodput within a Static Ad Hoc Environment. Proceedings of the 2006 international conference on Wireless communications and mobile computing. DOI= http://doi.acm.org/10.1145/1143549.1143650

[11] Singh, H., and Singh, S. 2002. Energy Consumption of TCP Reno, Newreno, and SACK in Multi-Hop Wireless Networks. Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems. DOI= http://doi.acm.org/10.1145/511334.511360

[12] Stemm, M., and Katz, R.H. 1997. Measuring and Reducing Energy Consumption of Network Interfaces in Hand-Held devices. IEICE Transactions on Communications, E80-B(8):1125-1131, 1997.