Profiling Power Consumption on Mobile Devices

Luca Ardito, Giuseppe Procaccianti, Marco Torchiano, Giuseppe Migliore
Dipartimento di Automatica ed Informatica
Politecnico di Torino
Torino, Italy
E-mail: name.surname@polito.it

***Abstract*—The proliferation of mobile devices, and the migration of the information access paradigm to mobile platforms, motivate studies of power consumption behaviors with the purpose of increasing the device battery life. The aim of this work is to profile the power consumption of a Samsung Galaxy I7500 and a Samsung Nexus S, in order to understand how such feature has evolved over the years. We performed two experiments: the first one measures consumption for a set of usage scenarios, which represent common daily user activities, while the second one analyzes a context-aware application with a known source code. The first experiment shows that the most recent device in terms of OS and hardware components shows significantly lower consumption than the least recent one. The second experiment shows that the impact of different configurations of the same application causes a different power consumption behavior on both smartphones. Our results show that hardware improvements and energy-aware software applications greatly impact the energy efficiency of mobile devices.**

Keywords-Software; Energy Aware; Energy Profiling; Power Consumption; Android;

I. INTRODUCTION

Modern handheld devices are increasing their diffusion sharply. This means that, inevitably, the paradigm for accessing information and Internet services is migrating to mobile. The computational capabilities of this class of devices are growing rapidly, while their size is decreasing and they are increasingly becoming part of our daily life by providing us a wide range of functionality. Many features of modern devices like high processor speed, more efficient displays, more powerful data storage, Wi-Fi/GPRS/UMTS network adapters, advanced 3D graphics, etc. enable users to connect the now pervasive network, allowing access to personal data, as well as public resources anytime, anywhere. However, the use of these features significantly affects the energy consumption of the portable device and the advances in the development of battery technologies cannot keep pace with the rapid growth of energy demand.

This issue moved technology producers, information systems managers, and researchers to deal with energy consumption reduction [1]. For this reason, research has increasingly focused on improving the energy efficiency of hardware, but the literature still lacks in quantifying accurately the energy impact of software. Software does not consume energy directly, however it has a direct influence on the energy consumption of the hardware underneath. The battery is thus apparently a key parameter to control, in order to manage the power budget of the mobile device. It is essential to have precise figures of the current energy consumption of mobile devices to understand how to reduce their power consumption and how to design future energy efficient equipment. Today these figures are incomplete and not precise. The aim of this work is profiling the energy consumption behavior of a Samsung Galaxy I7500 and of a Samsung Nexus S in order to understand how the typical energy consumption of smartphones varied over the years.

We designed this work in two parts: the first one aims at identifying different usage scenarios corresponding to “high-level features”, which do not take into account what is underneath. The second part profiled the power consumption of a known source code, which provides context-aware functionalities.

The main contributions of this paper are:

- A profiling test bench, which allows executing defined scenarios on mobile devices and profiling the related power consumption through external measurement hardware;
- A comparison between power consumption of two different generations of Android OS-based mobile devices validated by statistical analysis of gathered data.

The remainder of this paper is structured as follows: Section II introduces the related work, Section III describes the context of our work, including instrumentation and research questions, Section IV presents results while Section V discusses them and, finally, Section VI provides conclusions and future works.

II. RELATED WORK

The energy profiling of mobile devices is an active research stream, especially as regards mobile and embedded devices. The concept of energy-awareness is based upon a complete knowledge on how and where energy is consumed on a device. In [2], authors present a detailed analysis of power consumption in a mobile device, focusing on the hardware subsystems, through common and realistic usage scenarios. Results show that the GSM module and the display are the most power-consuming components: for example, a GSM phone call on OpenMoko Neo Freerunner, HTC Dream G1 and Google Nexus One consumes 1135 mW, 822 mW and 846 mW respectively.

Usually, an accurate power consumption analysis of mobile or embedded devices is component-based. However,

 ISBN: 978-1-61208-259-2

instantaneous information about discharge current and remaining battery capacity is not always available, because most devices do not have built-in sensors to collect these data. In [3], a technique called PowerBooter is proposed to build a battery-based model automatically. Authors motivate this decision by considering that different mobile devices of the same category show different power consumption, and a specific power consumption model for each device is difficult to obtain. Thus, instead of using external metering instrumentation to detect power consumption, only the internal battery voltage sensor is used, which is found across many modern smartphones.

From a software engineering point of view, most contributions are devoted in developing frameworks and tools for energy metering and profiling. Also in [3], authors propose an on-line power estimation tool called PowerTutor. It implements the PowerBooter model in order to profile power consumption of applications, basing upon their component usage. Another example, which makes use of external metering devices, is ANEPROF [4], which authors define as a real-measurement-based energy profiler able to reach function-level granularity. It is developed for Android OS based devices, thus it is aimed at profiling Java applications. It is based on JVM event profiling, using software probes to record runtime events and system calls. Authors had to address several design issues, such as overhead control and proper time synchronization. Power consumption profiling is made through correlation of real-time power measurements done by an external DAQ, connected to a ARM Computer-on-Module running Android 2.0. Authors also provide profiling data of four popular applications (Android Browser, Gmail, Facebook, YouTube). The accuracy of ANEPROF depends on the hardware meter used. Its CPU overhead is stated to be less than 5%. Finally, SEMO [5] is a smart energy monitoring system, developed for Android, which provides also application-level consumption monitoring. This system is composed of three components: an inspector, which monitors the information on the battery, warning users when the battery reaches a critical condition; a recorder, which basically logs the actual charge of the battery and the running applications, and an analyzer, which calculates the energy consumption rate for each application and ranks them according to it.

As we have shown in this section, several efforts have been made as regards energy profiling in mobile devices. However, these works differ greatly in terms of methodologies and formalisms used. Palit et al. in [6] propose an interesting framework for performing experiments to measure the energy cost of software applications on smartphones.

They define the concept of user-level test case γ_i as a pair $<$ *input; output* $>$ where the input is composed of an application setting α_i and a device configuration β_i, and the output is the energy cost θ_i, expressed as a custom metric depending on battery capacity and amount of current consumed. Formally: $\gamma_i =< \alpha_i, \beta_i; \theta_i >$

Authors describe also a typical workbench for experimentation, which is very similar to the one we used in this work. In this contribution, we will follow the methodology and approach used in our previous work [7], where we performed an analog experiment aimed at assessing the software impact over power consumption in Desktop computer systems.

III. Study Design

The aim of our research is to compare the impact of software usage on power consumption in two different Android OS-based mobile phones. For this purpose, we performed two experiments: the first one, called “training tools”, fixes a set of high level features in order to compare the power consumption between the two devices, while the second one, called “gLCB”, executes different profiles of a Context-Aware application and compares its power consumption on the two devices according to the selected profiles.

A. Variable selection

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Experiment 1: Training Tools. We aim at quantifying, in two different models of smartphone, the power consumption of hardware components, when performing daily activities for a common user. We selected two independent variables: the smartphone model (M) and the specific scenario (S). Each has been executed 30 times, with a fixed duration of 4 minutes per scenario. Our dependent variable is the consumed power (P). The scenarios are:

S0: Standby. This scenario provides the baseline for our analysis. During this scenario, there are no user applications in execution, and 2G and 3G connections are enabled.

S1: Phone call over 2G network. This scenario executes a phone call to a prefixed number, for a total duration of 4 minutes.

S2: Phone call over 3G network. Same as above, except that the call is made using the UMTS network.

S3: File download through Wi-Fi connection. In this scenario, the scheduled task launches a new thread, which downloads a remote file, the Ubuntu 11.10 disk image, up until the scheduled timeout (4 minutes).

S4: File download through 2G (EDGE/GPRS) connection. Same as above, except that the downloaded file is the Android SDK, which is smaller in size.

S5: File download through 3G (UMTS) connection. Same as above, except that the UMTS network is used.

S6: Localization activity through GPS. This scenario manages position updates. The task simply registers on location updates and reads the new values of latitude and longitude, up to the 4 minutes timeout.

S7: Scan for Bluetooth devices. In this scenario, a scan for Bluetooth devices is performed. The scan process lasts, according to specifications, 12 seconds in average. At the end of the scan procedure, the task simply restarts, up until the prefixed duration.

S8: CPU-intensive activity. The aim of this scenario is maintaining a high CPU workload while gathering power consumption data. For this purpose, repeated cryptography operations are performed, with a pool of 20 threads, each of them iterating the procedure 10 times.

S9: Playback of an audio file. This scenario plays an mp3 compressed audio file, 4.78 MB in size, played in loop up until the scheduled timeout.

S10: Active display with 50% Brightness. The aim of this scenario is assessing the impact of the active display over power consumption. This scenario is similar to S0, the only difference being that all radios (2G, 3G, Wi-Fi) and the SIM card were disabled.

Experiment 2: gLCB. We analyze the energetic behavior of an application, called gLCB, the Android porting of a Context-Awareness application developed by Telecom Italia Lab [9]. Basically, its purpose is to retrieve diverse context information (such as geographical location, Wi-Fi hotspots, Bluetooth devices, etc.) from a portable device, in order to send it to a remote Context Provider for the implementation of Context-Aware services to the end user.

gLCB is based on an event mechanism, that triggers the data upload only when a context change is detected. Depending on the usage profile chosen for the application, which can be one of the following: *VERY LOW, LOW, NORMAL, HIGH, AUTO, CUSTOM*, the data retrieving and upload ratio are adjusted, thus affecting the energy behavior of the application. In our experiment, each profile was set through a server application and data was collected during execution sessions of the gLCB application of the fixed duration of 60 minutes for each profile.

B. *Hypothesis Formulation*

We define our goal through the Goal-Question-Metric (GQM) approach. [10]. For Experiment 1 the goal is: "Analyze usage scenarios of two mobile devices for the purpose of assessing differences with respect to power consumption from the viewpoint of the System User in the context of mobile applications" while for Experiment 2 the goal is: "Analyze usage profiles of gLCB source code for the purpose of assessing differences with respect to power consumption from the viewpoint of the System User in the context of mobile applications".

1) *Experiment 1: Training Tools:*

- *Research Question 1.1: Do usage scenarios have the same power consumption?*
  $H_{0,1} : P_{0,1} = P_{1,1} = ... = P_{10,1}$
  $H_{0,2} : P_{0,2} = P_{1,2} = ... = P_{10,2}$
- *Research Question 1.2: Is the energy consumption the same among the devices?*
  $H_0: P_{i,1} = P_{i,2}$, $i \in [0, 10]$
  $H_A: P_{i,1} \neq P_{i,2}$, $i \in [0, 10]$

2) *Experiment 2: gLCB:*

- *Research Question 2.1: Does gLCB cause a variation of the devices power consumption?*
  $H_{0,1}$: $P1_{\text{with gLCB}} = P1_{\text{without gLCB}}$
  $H_{0,2}$: $P2_{\text{with gLCB}} = P2_{\text{without gLCB}}$
- *Research Question 2.2: Are there statistical differences between different user profiles?*
  $H_{0,1}$: $P_{1\text{ high}} = P_{1\text{ normal}} = P_{1\text{ low}} = P_{1\text{ verylow}}$
  $H_{0,2}$: $P_{2\text{ high}} = P_{2\text{ normal}} = P_{2\text{ low}} = P_{2\text{ verylow}}$
- *Research Question 2.3: Are there statistical differences between the behaviors of gLCB in different devices?*
  H_0: $P_{i,1} = P_{i,2}$, $i \in$ (high, normal, low, verylow)

The metric is Power consumption.

C. *Instrumentation and Experiment Design*

The selected usage scenarios have been implemented in Java code using the Android SDK. In order to obtain a statistically relevant data set, each scenario has a fixed execution time of 4 minutes, and each execution was repeated 30 times. This procedure was equally applied on each smartphone.

Hardware Instrumentation. The experiments were performed on two different models of smartphones: the "Galaxy i7500", first announced in April, 2009, which is the first model produced by Samsung based on Android OS; and the "Nexus S", first announced in December, 2010, produced by Google and Samsung. Their technical specifications are listed in the producer website. The power consumption data was acquired through a power metering architecture. The battery was removed from the devices, in order to avoid bias due to discharge and subsequent OS power saving procedures. The battery terminals were directly connected to a DC power supply, providing 5 V steadily. This value was chosen after different tests, that showed how lower values were not able to maintain the device operational during the most power consuming scenarios, because of the voltage drop on the shunt resistance. The DC power supply used is the TPS-2000D produced by Topward Electric Instruments Co. A Data Acquisition Board (DAQ), the DAQLite produced by Eagle Technology, was used to acquire the power consumption data. The DAQ was set to a sampling frequency of 350Hz, in order to produce an amount of data statistically relevant, but not prohibitive for subsequent computation.

Software Setup. In order to automate scenario execution in our experiments, a supporting software environment was developed, composed of two Android applications, a server side application and macro scripts, to be executed by the tool AutoHotKey3. The developed Android application allows enabling or disabling components, such as Bluetooth, GPS or Wi-Fi interface, in order to avoid bias during scenarios that do not use them. For our second experiment, another Android application has been developed to control the execution of gLCB, specifying an execution time and a usage profile. With this solution, we assessed how the execution of different profiles of the application affected the power consumption of the device. These applications communicate with a server machine, which is then connected to the DAQ via USB. The server application then launches a AutoHotKey script that performs the needed operations for data acquisition and logging.

 ISBN: 978-1-61208-259-2

D. Instrumentation and Experiment Design

The goal of data analysis is to apply appropriate statistical tests to reject the null hypothesis. As we expected, the collected power consumption values, for both smartphones, do not follow normal distribution. This was verified by means of the Shapiro-Wilk test, with a resulting p-value lower than 0.05. This is true for our first experiment, "Training tools" as well as for the second one, "gLCB". Thus, in order to verify our hypotheses, we used non-parametric versions of the Kruskal-Wallis and Wilcoxon-Mann-Whitney tests [8], to assess the statistical independence between the different scenarios and profiles evaluated during our experiments. Again, we will draw conclusions from our tests based on a significance level $\alpha = 0{:}05$, that is we accept a 5% risk of type I error – i.e. rejecting the null hypothesis when it is actually true.

E. Threats to validity

We will classify threats of experiment validity in two categories: internal threats, derived from our treatments and instrumentation, and external threats that regard the generalization of our work. A possible internal threat concerns the sampling frequency adopted by the DAQ, namely 350 Hz. We chose this frequency value for practical reasons, in order not to obtain a huge amount of data, which could not be computed in a reasonable time by our servers. However, this frequency, compared to the operational frequencies of the selected smartphones, could be seen as quite low. A more significant threat comes from the usage, in some of our scenarios, of different communication networks, which are characterized by an unpredictable behavior. This behavior may add bias to our measurements, introducing high data variability. For example, as regards the cellular network, power consumption could be affected by the following mechanism: the base station to which the mobile device is connected detects the signal power, and if the SINR is below or above a specific threshold it may negotiate a signal power increase or reduction to the device antenna. Finally, although is it not possible to generalize our results, because we performed our experiments on two specific models of smartphones, it is however possible to consider them representatives of category of devices with similar specifications.

IV. RESULTS

A. Preliminary Data Analysis

We present in Tables I, II, III, IV the following descriptive statistics about collected data. Tables report in this order: median (milliWatts), mean (milliWatts), standard deviation (σ), and variation coefficient (the standard deviation divided by the mean). Tables II, III contain descriptive statistics for each scenario of our "Training Tools" experiment, while Tables IV, V contain descriptive statistics for each profile of our "gLCB" experiment.

TABLE I EXPERIMENT "TRAINING TOOLS": SCENARIOS STATISTICS – SAMSUNG GALAXY I7500

Scenario	Median (mW)	Mean (mW)	Std.Dev.	Var.Co
2G Standby	8.663	17.840	65.763	3.686
3G Standby	8.663	27.248	97.592	3.581
2G Call	658.618	746.447	371.118	0.497
3G Call	957.803	988.069	97.313	0.098
WiFi Download	628.724	646.604	61.403	0.094
2G Download	669.467	784.099	742.696	0.947
3G Download	955.175	947.515	181.155	0.191
GPS	450.189	484.753	79.748	0.164
Bluetooth Scan	251.526	273.960	78.018	0.284
CPU-Intensive	606.923	608.708	38.442	0.063
Mp3 Audio	324.720	374.971	142.073	0.378
Display	386.252	408.754	81.598	0.199

TABLE II EXPERIMENT "TRAINING TOOLS": SCENARIOS STATISTICS – SAMSUNG NEXUS S

Scenario	Median (mW)	Mean (mW)	Std.Dev.	Var.Co
2G Standby	8.663	26.830	55.572	2.071
3G Standby	8.663	18.958	48.708	2.569
2G Call	379.488	543.487	565.230	1.040
3G Call	846.688	878.850	126.708	0.144
WiFi Download	455.733	513.046	166.444	0.324
2G Download	605.874	722.422	854.086	1.182
3G Download	965.368	931.798	208.819	0.224
GPS	296.626	300.444	20.375	0.067
Bluetooth Scan	217.571	227.051	42.882	0.188
CPU-Intensive	886.552	877.747	54.055	0.061
Mp3 Audio	155.035	164.709	26.666	0.161
Display	598.734	708.075	177.169	0.250

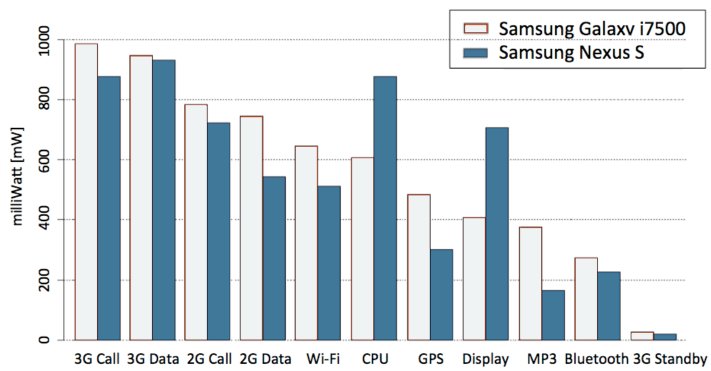

Figure 1 Instant Power Consumption (avg) comparison

TABLE III EXPERIMENT "GLCB": PROFILE STATISTICS - SAMSUNG GALAXY I7500

Scenario	Median (mW)	Mean (mW)	Std.Dev.	Var.Co
Very Low	396.474	521.565	213.243	0.408
Low	396.474	544.451	226.976	0.416
Normal	455.869	594.340	248.253	0.417
High	514.952	617.016	253.448	0.410

TABLE IV EXPERIMENT "GLCB": PROFILE STATISTICS - SAMSUNG NEXUS S

Scenario	Median (mW)	Mean (mW)	Std.Dev.	Var.Co
Very Low	420.271	536.863	225.055	0.419
Low	435.116	556.254	244.286	0.439
Normal	690.392	834.233	315.046	0.377
High	808.711	876.160	334.903	0.382

 ISBN: 978-1-61208-259-2

B. *Hypothesis Testing*

In this section we provide the results of hypothesis testing for our research questions. All p-values have been verified to be lower than the chosen significance level $\alpha = 0{:}05$.

- RQ 1.1: *Do usage scenarios have the same power consumption?*
  - $H_{0,1} : P_{0,1} \neq P_{1,1} \neq ... \neq P_{10,1}$
    Our values range from an average of 17.8 mW for Scenario S0 to an average of 988 mW for Scenario S2. The Kruskal-Wallis test for the hypothesis resulted in a p-value lower than 2.2e-16. Thus, we reject the null hypothesis.
  - $H_{0,2} : P_{0,2} \neq P_{1,2} = ... \neq P_{10,2}$
    Our values range from an average of 18 mW for Scenario S0 to an average of 931.8 mW for Scenario S5. The Kruskal-Wallis test for the hypothesis resulted in a p-value lower than 2.2e-16. Thus, we reject the null hypothesis.
- *RQ 1.2: Is the energy consumption the same among the devices?*
  - H_0: $P_{i,1} \neq P_{i,2}$, $i \in [0, 10]$
    The Mann-Whitney test resulted in a p-value lower than 0.001 for each scenario. Thus, we reject the null hypothesis.
- RQ 2.1: *Does gLCB cause a variation of the devices power consumption?*
  - $H_{1,1}$: $P_{1\ \text{with gLCB}} \neq P_{1\ \text{without gLCB}}$
    The Mann-Whitney test resulted in a p-value lower than 2.4e-09 for each profile compared to the standby consumption. Thus, we reject the null hypothesis.
  - $H_{1,2}$: $P_{2\ \text{with gLCB}} \neq P_{2\ \text{without gLCB}}$
    The Mann-Whitney test resulted in a p-value lower than 1.5e-09 for each profile compared to the standby consumption. Thus, we reject the null hypothesis.
- *RQ 2.2: Are there statistical differences between different user profiles?*
  - $H_{2,1}$: $P_{1\ \text{high}} \neq P_{1\ \text{normal}} \neq P_{1\ \text{low}} \neq P_{1\ \text{verylow}}$
    Our values range from an average of 521.5 mW for Very Low profile to an average of 617 mW for High profile. The Kruskal-Wallis test for the hypothesis resulted in a p-value lower than 1.146e-15. Thus, we reject the null hypothesis.
  - $H_{2,2}$: $P_{2\ \text{high}} \neq P_{2\ \text{normal}} \neq P_{2\ \text{low}} \neq P_{2\ \text{verylow}}$
    Our values range from an average of 536.8 mW for Very Low profile to an average of 876 mW for High profile. The Kruskal-Wallis test for the hypothesis resulted in a p-value lower than 3.433e-15. Thus, we reject the null hypothesis.
- *RQ 2.3: Are there statistical differences between the behaviors of gLCB in different devices?*
  - $H_{2,1}$: $P_{1\ \text{high}} \neq P_{1\ \text{normal}} \neq P_{1\ \text{low}} \neq P_{1\ \text{verylow}}$
    Our values range from an average of 521.5 mW

The Mann-Whitney test resulted in a p-value lower than 2e-10 for each profile. Thus, we reject the null hypothesis.

The bar plot in Figure 1 shows the average power consumption values in mW for each scenario, on both smartphone models.

As we expected, we may notice that the standby values are the lowest, below 27 mW in the worst case. From the graph, it is evident how the most recent smartphone, the Samsung Nexus S, consumes a significantly lower amount of power in each scenario, exception given by the CPU Intensive and the Active Display scenarios. The percentage variations between the two smartphones, reported in Table VI, spread from a minimum -56,08% in the Mp3 Audio scenario, to a +73,23% in the Active Display scenario.

As regards our gLCB experiment, the bar plot in Figure 2 shows the average power consumption values in mW for each profile, on both smartphone models. It is possible to notice that, between the two, the most recent smartphone consumes less in verylow and low profiles.

V. DISCUSSION

A. *Experiment 1: "Training Tools"*

From the results obtained from our first experiment, we can conclude that the most power consuming user activities, among the ones we selected, on both the smartphone models used, are those that use the radio module, namely the phone

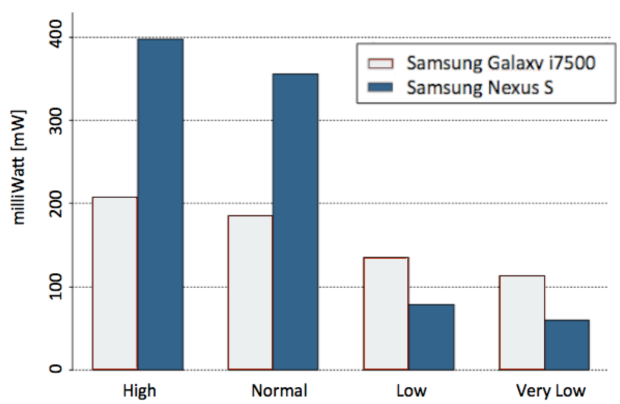

Figure 2 Instant Power Consumption (avg) of gLCB energy profiles

TABLE V COMPARATION IN MILLIWATTS OF SCENARIOS POWER CONSUMPTION OF THE TWO SMARTPHONES

Scenario	Galaxy	Nexus S	Galaxy VS Nexus S
2G Display	408.750	708.080	+73,23%(299.330 mW)
CPU Intensive	608.710	877.750	+44,19% (269.040 mW)
3G Download	947.510	931.800	-1,65% (15.710 mW)
2G Download	784.100	722.420	-7,86% (61.680 mW)
3G Call	988.060	878.850	-11,05% (109.210 mW)
Bluetooth Scan	273.960	227.050	-17,12% (46.910 mW)
Wi-Fi	646.600	513.050	-20,65% (133.550 mW)
2G Call	746.450	543.490	-27,19% (202.960 mW)
GPS	484.750	300.450	-38,02% (184.300 mW)
Mp3 Audio	374.980	164.700	-56,08% (210.280 mW)

 ISBN: 978-1-61208-259-2

TABLE VI COMPARATION IN MILLIWATTS OF PROFILE POWER CONSUMPTION (WITHOUT DISPLAY OVERHEAD)

Profile	Galaxy	Nexus S	Galaxy VS Nexus S
Very Low	112.811	59.185	-47,54 % (53.626 mW)
Low	135.697	78.576	-42,09 % (57.121 mW)
Normal	185.586	356.555	+92,12 % (170.969 mW)
High	208.262	398.482	+91,34 % (190.22 mW)

call and data transfer on both 2G (EDGE/GPRS) and 3G (UMTS) networks. This finding is coherent with [2], in which the most power-consuming scenario is indeed the phone call. Also in [6], examining some of the network related scenarios, it emerges that the battery lasts longer in all cases if the WiFi network interface is used, rather than 2G or 3G. From our results, it is worth to notice how 3G networks causes a sensible increase in power consumption with respect to 2G, in both voice and data communications.

Moreover, as regards the power consumption difference between the two smartphone models, we notice that the most recent model has in general a lower power consumption, the only exception being the CPU-intensive and active display scenarios: this can be justified, considering the increase in CPU frequency (1 GHz compared to 528 MHz) and in display dimensions (4” compared to 3.2”), which characterize the most recent model.

B. *Experiment 2: “gLCB”*

As regards our gLCB experiment, it is immediate to notice that in every profile the Samsung Nexus S consumes a higher amount of power. This is likely because during this experiment, the display of the smartphone was active, in order to verify the correct execution of the application, and the switching between profiles and execution sessions. This was done on purpose, in order not to introduce random bias due to occasional checking of the application behavior. Instead, we may subtract the display overhead from the power consumption values, and this is valid because we know that, from the previous experiment, the display consumption is characterized by low variance and dispersion values. The recalculated values, without the display overhead, are shown in Table VII. It is interesting to notice how, from these results, emerges that the Nexus S has actually lower power consumption values than the Galaxy in profiles verylow and low, namely 47,5% and 42,1% respectively. The other profiles show significantly higher power consumption. Given that the step-up from low to normal profile is characterized by the activation of WiFi, Bluetooth and GPS components, during normal and high profiles a higher computational load is expected. Thus, we may conclude that the power consumption increase is due to the CPU activating more frequently than in the other two profiles, also because we know, from the results of the previous experiment, that the CPU has a higher impact on the Nexus S smartphone. These results show that the impact of the gLCB application in terms of power consumption gradually reduces, by adopting lower energy profiles, on both smartphones.

VI. CONCLUSIONS AND FUTURE WORKS

From the analysis of the results provided by our experiments, we can conclude that the most recent device, in terms of OS and hardware components, shows significantly lower power consumptions than the least recent one, except for the CPU-intensive and active display cases. We showed that differences of energy-awareness, based upon a complete knowledge on execution profiles of the same application, can significantly affect the power consumption of a device.

This finding shows that energy-aware software applications can improve the energy efficiency of mobile devices, while providing the same functionalities.

As regards future works, it would be interesting to profile the energy consumption of other usage scenarios, for example those who require a higher interaction between the user and the device. Moreover, because of our experiment design, the smartphones were constrained to a single physical location; it could be interesting profiling the power consumption of a moving user, in order to get closer to the real case and evaluate with more precision the contribute of the subsequent handoffs between different cells in mobile networks. Another interesting point of view could be analyzing the power consumption of different generations of smartphones running the same version of the Android OS, in order to isolate the only impact of hardware changes.

REFERENCES

[1] A. Berl, E. Gelenbe, M. Di Girolamo, G. Giuliani, H. De Meer, M. Q. Dang, and K. Pentikousis, “Energy-efficient cloud computing,” The Computer Journal, vol. 53, no. 7, pp. 1045–1051, 2009.

[2] A. Carroll and G. Heiser, “An analysis of power consumption In a smartphone,” in Usenix technical conference, Boston, MA, USA,, pp. 1–14, 2010.

[3] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. P. Dick, Z. M. Mao, and L. Yang, “Accurate online power estimation and automatic battery behavior based power model generation for smartphones,” in 8th IEEE International conference on HW/SW codesign and system synthesis, pp. 105–114, 2010.

[4] Y.-F. Chung, C.-Y. Lin, and C.-T. King, “Aneprof: Energy profiling for android java virtual machine and applications,” Parallel and Distributed Systems, International Conference on, vol. 0, pp. 372–379, 2011.

[5] F. Ding, F. Xia, W. Zhang, X. Zhao, and C. Ma, “Monitoring energy consumption of smartphones,” CoRR, vol. abs/1201.0218, pp. 610-613, 2012.

[6] R. Palit, R. Arya, K. Naik, and A. Singh, “Selection and execution of user level test cases for energy cost evaluation of smartphones,” in Proceedings of the 6th International Workshop on Automation of Software Test, pp. 84–90, 2011.

[7] G. Procaccianti, L. Ardito, A. Vetro’, and M. Morisio, “Profiling power consumption on desktop computer systems,” LECTURE NOTES IN COMPUTER SCIENCE, vol. 6868, pp. 110–123, 2011.

[8] L. Sachs, “Applied Statistics--A Handbook of Techniques”, Springer-Verlag, 1984.

[9] L.Ardito, M.Torchiano, M. Marengo, P. Falcarin, gLCB: An Energy Aware Context Broker, Sustainable Computing: Informatics andSystems, ISSN 2210-5379, pp. 1-9, 2013.

[10] R. Van Solingen and E. Berghout, Goal/Question/Metric Method. McGraw-Hill Inc.,US, January 1999.