



HAL
open science

An Experimental Methodology for Modeling the Energy Consumption of Mobile Devices

Khalil Ibrahim Hamzaoui, Mohammed Berrajaa, Mostafa Azizi, Giuseppe Lipari, Pierre Boulet

► **To cite this version:**

Khalil Ibrahim Hamzaoui, Mohammed Berrajaa, Mostafa Azizi, Giuseppe Lipari, Pierre Boulet. An Experimental Methodology for Modeling the Energy Consumption of Mobile Devices. First international conference on Embedded & Distributed Systems, EDiS'2017, Dec 2017, Oran, Algeria. hal-01723503

HAL Id: hal-01723503

<https://hal.science/hal-01723503v1>

Submitted on 5 Mar 2018

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

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

An Experimental Methodology for Modeling the Energy Consumption of Mobile Devices

Khalil Ibrahim HAMZAOUI Mohammed BERRAJAA Mostafa AZIZI Giuseppe LIPARI
Univ. Lille, CNRS, Centrale Lille, Univ. Mohammed Premier, Univ. Mohammed Premier, Univ. Lille, CNRS, Centrale Lille,
UMR 9189 – CRIStAL – Centre FSO, Lab. LANOL ESTO, Lab. MATSI UMR 9189 – CRIStAL – Centre
de Recherche en Informatique, 60000, Oujda, Morocco 60000, Oujda, Morocco de Recherche en Informatique,
Signal et Automatique de Lille, berrajaamo@yahoo.fr azizi.mos@ump.ma Signal et Automatique de Lille,
F-59000, Lille, France F-59000, Lille, France
Univ. Mohammed Premier, giuseppe.lipari@univ-lille.fr
FSO, Lab. LANOL
60000, Oujda, Morocco
hamzaoui.khalil@gmail.com

Pierre BOULET
Univ. Lille, CNRS, Centrale Lille,
UMR 9189 – CRIStAL – Centre
de Recherche en Informatique,
Signal et Automatique de Lille,
F-59000, Lille, France
pierre.boulet@univ-lille.fr

Abstract—Energy consumption is the result of interactions between hardware, software, users, and the application environment. Optimization of energy consumption has become crucial, energy metric is considered a critical metric, so it is important knowing how to measure and understand how energy is consumed on mobile devices. Accurate knowledge will allow us to propose different solutions to reduce energy consumption in order to improve the user experience. In this paper we propose an experimental methodology to build a model of the energy consumption of an application. We show in this paper how to build a simple predictive model of the energy consumption of an unconnected application based on precise measurements.

Index Terms—Mobile computing, Operating system, Energy consumption modeling.

I. INTRODUCTION

The energy consumption of a smartphone is the amount of energy used to operate its services [7]. The energy consumed by modern applications increases with each new version. Moreover, hardware components used by the applications define the resulting energy consumption. To know how to measure and to understand how the energy is consumed on these mobile devices remains an important objective of every developer [6].

The research presented in this paper is the development of an experimental methodology for modeling the energy consumption in mobile environments and more specifically in

Android platforms. The main objective of this work is to model the energy consumption of a particular application running on a mobile device.

In this paper we propose a model, we also describe a methodology to identify the parameters of the model (processor frequency, dissipated power and initial battery Level). To this end, we analyzed a collection of experimental data collected during a “Tour de France” in a wheelchair.

Then we applied statistical tools to obtain the parameters of the model. We finally validate the model by comparing results against other experimental data.

We present the methodology in section II, and we validate this methodology on experimental data in section III. Then we present related works in section IV and conclude.

II. METHODOLOGY

The research presented proposes a methodology that allows the modeling and evaluation energy costs in mobile environments. The aim of this work is to monitor energy consumption by acting on the following parameters:

- Frequency of processors,
- Initial level of the battery,
- Energy dissipated by clock frequency.

The model can be used to define an optimal frequency in terms of energy consumption for precise situations without degrading too much the quality of service desired by the user.

Collection of scientific data was carried out on the basis of several states on different places by fixing metrics previ-

ously. These measures comparisons were carried out in a real environment and confronted with the consumption measures carried out beforehand in a controlled environment.

The tools used to develop our study are:

Trepp Profiler Trepp Profiler is a diagnostic tool for profiling performance and power consumption of Android applications. All tests of this experimentation were processed by version V6.2s.

Trepp profiler provides information on system status, network status, graph performance, speed, processor frequency etc.

Cronoid Cronoid is an automation tool that allows performing tasks on a regular bases (like cron). It also makes it possible to carry out tasks automatically when the status of the terminal has changed. The version used for this work is Cronoid-3.5.1.

CPU Frequency CPU Frequency is a tool that allows you to change the CPU frequency setting to save energy or achieve better performance.

Our methodology is based on the following steps:

- Preparation of the test platform (CPU frequency management based on the governor) in order to have the rights to fix the frequencies of the CPU. For the fixation of the frequencies, we used the tool CpuFrequency.
- The role of the Cronoid tool is to automate tasks in order to minimize interaction with the user.

For the realization of our methodology, we based ourselves on the following information obtained by TreppProfiler:

- Total load per CPU,
- Memory usage,
- CPU frequency,
- Battery level,
- Battery power.

All these measures are regularly recorded in a table at a fixed measuring interval.

The methodology used to identify the parameters of the model is based on the determination of a correlation between the number of total operations Op , the dissipated elementary energy E under a previously set frequency F which will be expressed as $E(F, Op)$.

The total number of operations, Op , is calculated by summing the active cycles of each processor per measurement interval. This active cycles number is the load of the CPU multiplied by the measurement interval duration and by the CPU frequency. The dissipated energy during a measurement interval is the product of the battery power and the measurement interval duration.

The first step of the methodology is to compute a regression of the energy per measurement interval as a function of Op separately for some fixed (using CpuFrequency) frequencies. In our case study, a linear regression is enough to have a good model. Then we compute a regression of the coefficient of the fixed frequency models to obtain a model, $E(F, Op)$, of the energy consumption per measurement interval in function of the frequency F and the number of operations during this

measurement interval. In our case study, once again, a linear regression is enough to build a good model.

If one cannot or does not want to fix the CPU frequency, the same methodology can be applied on a variable frequency experiment by filtering the rows of the recorded measurement table by frequency and dealing separately with each frequency. The measuring intervals when the frequency change can simply be discarded.

III. MEASUREMENTS

A. Experimental context

Each service activated in smartphone induces an energy consumption relative to an amount of energy dissipated.

The purpose of this study is to provide a detailed follow-up on the main source's of energy consumption. We have used different frequencies. In the rest of our work, we will use the GSM network, which will be the only one to be in active state. The other communication networks (3G, 4G, Wi-Fi and the GPS) remain inactive throughout the various experimentations.

Measures related to this work were collected with a tour of France in electric wheelchair for 33 days and 3006 km.

Each experiment was carried out over a period of 45 minutes with a rate of ten measurements per second, which is worth 27000 measurements per test case. For the scenario Local Video with Fixed Frequency (LVFF), twenty experiments were carried out with different frequencies, battery level and locations totaling 540000 measurements.

In order to carry out our study, we relied on an electric wheelchair equipped with several sensors (speed, acceleration, position etc.) and a tablet (Hp pro slate8) equipped with Screen Features: 10-point multi-touch, technology Direct Bonding, Quad-Core, Processor type QUALCOMM Snapdragon 800 2.3 GHz, Storage memory 32 GB eMMC, RAM 2 GB LPDDR3 SDRAM, Lithium-polymer battery technology 21 Wh with a running time that can reach up to 13 hours and 45 minutes in case of video playback and Android system (version 4.4.4 - KitKat).

B. Modeling of energy consumption

In this section we will study the case of a local video with the fixed frequency in a motion state to test the impact of the variation of the GSM network on the energy consumption in disconnected mode.

Disconnected mode disables all linked sources to the connection such as Wi-Fi, Bluetooth and GPS. In this mode based on local data for energy modeling, only the GSM network will be active.

First scenario (LVFF) Local video with fixed frequency: in this scenario the CPU is used to monitor the system state by initiating and measuring the overall system power consumption for a local video with fixed frequency in advance.

Second scenario (LVVF) Local video with variable frequency: in this scenario the CPU is used to monitor the system state by initiating and measuring the overall system power consumption for a local video with variable frequency.

This scenario is used for the validation of the proposed methodology trained with the scenario with fixed frequencies.

The existing applications during the measurements are:

Facebook, GsmService, Google App, Clock, Avast Mobile Security, Cronoid, CPU Frequency, Google Play Store, Launcher3, Power Battery, Google services, Play, Hangouts, Multimedia storage, Google Play Music, Play-Fi, Keyboard, Google, MusicFX, SmartcardService, System interface, Messenger, VLC, Cron Tasker Free, and Google Partner Configuration.

In all the experiments of this scenario, we based ourselves on a video of 720×302 resolution with 25000 frames using the MPEG-4 video codec over a period of 45 min, the same experiment has been repeated with different processor frequencies (previously set) and with different battery levels.

The aim of this part is to evaluate the dissipated energy per clock cycle which will be called “dissipated elementary energy”. Clock Cycle is the speed of a computer processor which is the amount of time between two pulses of an oscillator. The clock speed is measured in Hz.

To qualify the quality of the model, it was based on the value of the coefficient of determination which represents the square of the linear correlation coefficient R which is calculated as follows:

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \in [0, 1]$$

y_i are the values of the measurements, \hat{y}_i the predicted values and \bar{y} the mean of the measurements.

The average consumption per clock can be expressed as follows:

$$E_{\text{clock}} = \text{Regression} \left(\sum_{i=1}^4 \text{Load}_i, E_{\text{dissipated}} \right)$$

E_{clock} shows the total energy per clock, Load_i the load of the CPU $_i$ and $E_{\text{dissipated}}$ corresponds to the dissipated energy measured experimentally.

C. Results of measurements

The various measures relating to the energy dissipated were carried out in relation to the number of total operations at a fixed frequency. The figure 1 shows an example of experimentation by fixing the frequency of the processor at 1.5 GHz.

The graphs corresponding to the other frequencies (1 GHz, 1.7 GHz and 2.2 GHz) are presented in the same way, they are presented in the figure 2.

The respective equations of the results obtained above are:

$$E(1.0, Op) = 0.0512 + 5.995.e^{-11}Op, R^2 = 0.830$$

$$E(1.5, Op) = 0.0321 + 4.226.e^{-11}Op, R^2 = 1.000$$

$$E(1.7, Op) = 0.0337 + 5.073.e^{-11}Op, R^2 = 1.000$$

$$E(2.2, Op) = 0.0309 + 7.210.e^{-11}Op, R^2 = 1.000$$

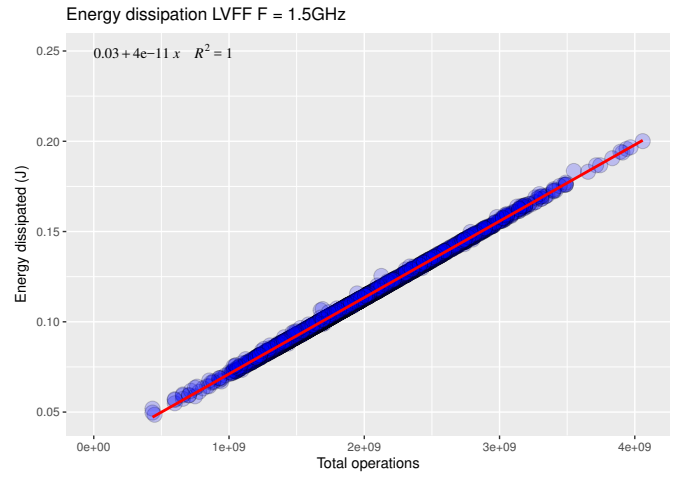


Fig. 1. Energy dissipation for LVFF model, Frequency=1.5GHz

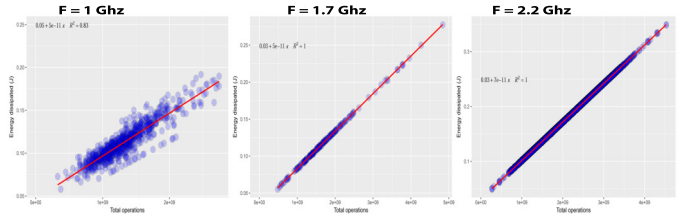


Fig. 2. LVFF frequencies = 1 GHz, 1.7 GHz, 2.2 GHz

The value of the determination coefficient is equal to 1 in the case of measurement with the frequencies 1.5 GHz, 1.7 GHz and 2.2 GHz, which proves the efficiency of the proposed methodology for the local video. In the case of a frequency fixed at 1 GHz, the coefficient of determination is equal to 0.830. The relevance of the linearity hypothesis was affected by the beginning of degradation of the quality of the video (following the frequency imposed in the experimentation) which explains the result obtained.

On the basis of the experimental results obtained, we can deduce the energy discharge per cycle while playing a local video with a fixed frequency. The dynamic consumption corresponds to the energy consumption of the processor due to the fact that the transistors change state. The dynamic energy consumption per cycle for the measured frequencies is presented in figure 3.

The static consumption corresponds to the energy dissipated when the transistors remain in the same state. The static energy consumption for the measured frequencies is presented in figure 4.

The respective equations of dynamic energy and static energy dissipated for our model obtained by linear regression are:

$$E_{\text{Dynamic}}(F) = 4.008.e^{-11} \times F - 2.101.e^{-11}, R^2 = 1.000$$

$$E_{\text{Static}}(F) = 0.03 \times F - 0,009, R^2 = 1.000$$

From these measurements we propose the following model

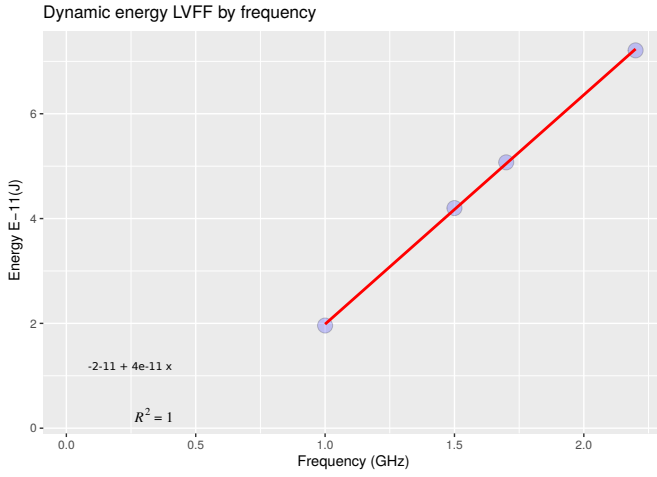


Fig. 3. Dynamic energy dissipated by clock in function of the frequency

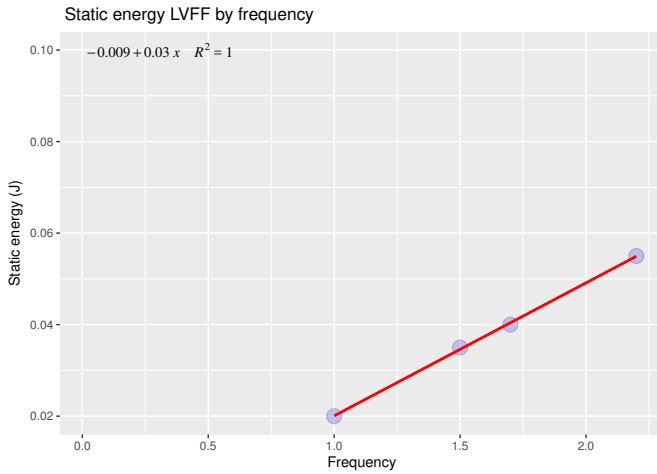


Fig. 4. Static energy in function of the frequency

$$E(F, Op) = (A \times F + b) \times Op + (C \times F + d)$$

Where E is the total energy, F the selected frequency, Op is the number of operations per measurement interval (in our case 100 ms), $A \times F + b$ is the dynamic energy consumption, and $C \times F + d$ is the static energy.

For the case of our study, the coefficients related to the LVFF scenario are thus defined by the following equation:

$$E(F, Op) = (4.008.e^{-11} \times F - 2.101.e^{-11}) \times Op + (0.03 \times F - 0,009)$$

D. Validation of mathematical modeling

In order to verify our model, we use an experiment that has not been used for the development of the model, we have the results relative to the dissipated energy (measured experimentally) and we compare them with the energy consumption predicted by the proposed LVFF model.

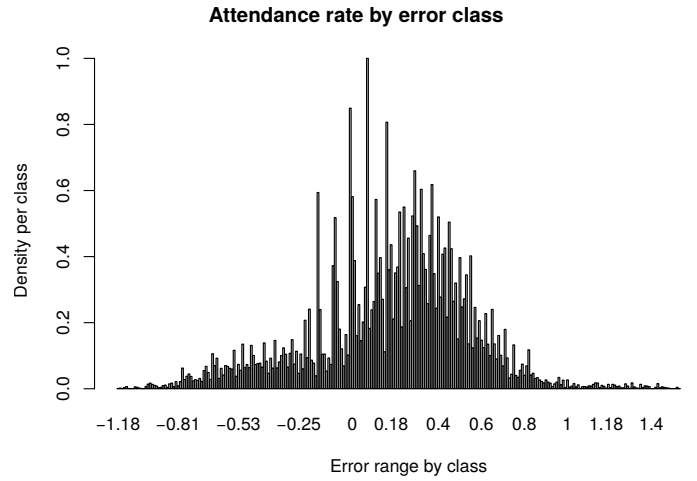


Fig. 5. Normalized error distribution using 1000 error class

The experiments that will be used for the test consist of 45,945 measurements with a variable frequency. Thus for each measurement interval, we use the recorded frequency as input to the model. The comparison of the predicted quantities by the above mathematical model and the measurements obtained experimentally make it possible to define the precision of the predicted model. After comparison, there is a strong coherence with a relatively low error rate.

The average relative error between our model and the experimental measurements corresponds is 1.767 % with a standard deviation of 3.652 %. Further more, 93.9 % of the measurements obtained have an error rate of less than 8 % and 64.73 % of the measurements obtained have an error rate of less than 5 %.

To illustrate this coherency, we have plotted the distribution of the cumulative sum, for this purpose, a comparison is made with the measurements obtained experimentally and that obtained by our model, the sum of the errors obtained in the various measurements was made in order to evaluate the cumulative error rate in all the experimentation. The error rates are grouped by error classes (we have chosen 1000 classes) in the figure 5. The vertical axis was standardized according to the number of existing classes, 1 corresponds to the largest error class (the one with the highest number of members).

The results obtained correspond to a Gaussian distribution, the class which contains the greatest number of errors corresponds at the error interval 0.1 %, a value which consolidates the choice of parameters of model LVFF.

IV. COMPARISON WITH THE STATE OF THE ART

A. TailEnder

Balasubramanian, Balasubramanian and Venkataramani designed TailEnder [1] for measuring the characteristics of the energy consumption of the various smartphones. TailEnder aims to minimize energy consumption for applications that can tolerate delay such as e-mails. In particular, the study of 3G's

energy consumption characteristics important and non-intuitive implications for the design of efficient energy application. Analytical modeling: Previous studies such as [2], [8], [12] have studied the impact of different energy-saving techniques in 3G networks using analytical models [7].

B. Cinder

Roy, Rumble, Stutsman, Levis, Mazieres and Zeldovich show in [11] that the Cinder method, that is designed for phones and mobile devices, allows users and applications to control and manage the limited resources of the device such as energy.

Cinder's mechanism [11] consists of detecting the main energy-consuming tasks and then assigning its information to the guiding applications in order to allocate the energy resources to the priority tasks [3], [4], [14], [15].

C. Eprof

Pathak, Hu and Zhang have modeled Eprof [9], a good part of the components such as Wi-Fi, 3G GPS etc has a behavior of the tail [10] in which a component can enter a state of high power and remain in that state of energy beyond the end of the route.

This approach evaluates the energy consumed by the I/O and the energy of the tail [1], [10] as well as the energy consumed because of wake-up caused by the wake locks.

D. RAPL

Hahnel, Dobel, Volp and Hartig, developed the approach RAPL (Running Average Power Limit) that allows to compare applications of two sectors and shows that they have different energy consumption while offering similar services [5].

Running Average Power Limit (RAPL) is based on energy sensors available for measuring the energy consumption, RAPL takes into account the energy consumption in the software components.

The aim of RAPL methodology is to measure the energy consumed by the device and that consumed by its pilot.

E. AppScope

AppScope is a system that automatically evaluates energy applications running on Android smartphones' consumption [13]. Its design is based on monitoring the Android kernel at a microscopic level. The objective of AppScope model is to measure the energy application system for applications that uses power equipment models and usage statistics for each hardware component

We have previously published a more detailed study on the models of the state of art in [7].

Table I compare these energy consumption methodologies and the one we propose in this article on different criteria.

Meaning of the columns:

Quality model: quality of the energy model, this criterion shows the size of the work devoted to providing an energy model.

TABLE I
COMPARISON OF ENERGY CONSUMPTION METHODOLOGIES

	Quality model	Optim. hard	Optim. appli.	QoS	Sécurité	Dev.
TailEnder	+	++	NA	+	NA	NA
Cinder	+	++	NA	++	+++	NA
Eprof	++	+	+	+	NA	++
RAPL	+	++	NA	+	+	+
AppScope	+	NA	+++	+	NA	+
LVFF	++	+	++	+	NA	+

Optim. hard: taking into account hardware optimizations, this criterion shows the quality of the energy model and the material energy consumption.

Optim. application: taking into account software optimizations, shows the quality of the energy model according to the implementation of software.

QoS: impact on quality of service, Quality of Service (QoS) is a management concept that consists of giving priorities to a few applications to keep a desired level of performance. This criterion details the effect on QoS during the reduction of energy consumption.

Security: taking security into account, The security criterion role is to prevent malware to reducing battery life.

Dev: developer help, this criterion is used to reduce the energy consumption of this software when designing.

Eprof and our model LVFF achieve better results for quality of the energy model and performance optimization. Eprof makes it possible to analyze the state of the energy demands and to estimate the rate of tail energy of some components. Our model LVFF allows to model the energy consumption so that it is optimized for better performances.

TailEnder, Cinder, RAPL satisfy the Optim hard criterion. TailEnder performs a minimization of the energy consumption for the application which tolerate a small delay by quantifying the use of the energy, Cinder by allocating resources to the guiding applications by visualizing the rate of limited resources. RAPL makes an energetic comparison of a similar service and shows that the consumption is not identical.

Appscope and our model give the possibility of optimization of existing applications. Appscope evaluates automatically the power consumption of applications at a microscopic level. It allows to detect events relevant to the operation of a component for better optimization. Our model LVFF allows to define the optimal parameters for a better performance which justifies their relevance in the criterion Optim Application.

The quality of service is well managed by Cinder because it automatically manages the allocation of energy resources, the LVFF model does not have much impact on quality of service because user interaction is minimal due to automation of scenario tasks with the cronoid tool.

Cinder worked well the security system that manages the emergency call in case of scarcity of energy resources.

Eprof satisfies the Dev criterion because it can help the developers as it can analyze the different energy states of the

device (characteristics, tail energy ect.)

F. Advantages of the proposed model

- Our model is based on the most elementary entity which allows a high degree of precision for the evaluation of the results.
- The identification of the model parameters is relevant because it is based on measurements carried out in a real and diversified environment.
- The proposed methodology allows for optimization by tracking sources of energy consumption.
- The proposed methodology can be used for decision support for a desired performance.

G. Limits of the proposed model

- The precision of the model depends on the electrical characteristics of the device.
- The proposed methodology is not automated, it requires manual installation of the tools used to developing the model.
- The regression type is not automatically detected.

V. CONCLUSION

We have presented in this paper a methodology to build a model of the energy consumption of applications on mobile devices. This methodology starts by recording accurate measures of the energy consumption during the use of selected applications. Using these measures, we build a model of the energy consumption in function of the activity of the processing cores for fixed frequencies by (linear) regression. The last step in our methodology consists in building a model of the energy consumption in function of the operating frequency and the activity of the processor by a regression on the parameters of the fixed frequency models. We have demonstrated this methodology on an experimental test case over local video playing. As this test case does not involve communication, the prediction capabilities of the model are quite good. The main quality of our methodology regarding related works is the relevance of the recorded data that leads to an accurate model. The proposed solution can be used to define an optimal frequency for one or more applications in order to offer a better user experience with a reduced energy consumption.

As future work, we will extend this methodology to deal with communicating test cases. We expect the model to be less accurate due to the variability of the communication patterns and of the connectivity of the wireless link. Such a methodology can help developers to understand the energy consumption behavior of their applications while optimizing the quality of service.

REFERENCES

- [1] Niranjan Balasubramanian, Aruna Balasubramanian, and Arun Venkataramani. Energy consumption in mobile phones: a measurement study and implications for network applications. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, pages 280–293. ACM, 2009.
- [2] M Chuah, Wei Luo, and Xiaowei Zhang. Impacts of inactivity timer values on umts system capacity. In *Wireless Communications and Networking Conference, 2002. WCNC2002. 2002 IEEE*, volume 2, pages 897–903. IEEE, 2002.
- [3] Jason Flinn and Mahadev Satyanarayanan. Powerscope: A tool for profiling the energy usage of mobile applications. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on*, pages 2–10. IEEE, 1999.
- [4] Rodrigo Fonseca, Prabal Dutta, Philip Levis, and Ion Stoica. Quanto: Tracking energy in networked embedded systems. In *OSDI*, volume 8, pages 323–338, 2008.
- [5] Marcus Hähnel, Björn Döbel, Marcus Völp, and Hermann Härtig. Measuring energy consumption for short code paths using rapl. *ACM SIGMETRICS Performance Evaluation Review*, 40(3):13–17, 2012.
- [6] Khalil Ibrahim Hamzaoui, Wiame Benzekri, Gilles Grimaud, Mohammed Berrajaa, and Mostafa Azizi. Estimation and optimization of energy consumption on smartphones. In *Proceedings of the Mediterranean Conference on Information & Communication Technologies 2015*, pages 333–342. Springer, 2016.
- [7] Khalil Ibrahim Hamzaoui, Gilles Grimaud, Mostafa Azizi, Mohammed Berrajaa, and Abdelkader Betari. Survey on adaptation techniques of energy consumption within a smartphone. In *Science and Information Conference (SAI), 2014*, pages 247–253. IEEE, 2014.
- [8] Chi-Chen Lee, Jul-Hung Yeh, and Jyh-Cheng Chen. Impact of inactivity timer on energy consumption in wcdma and cdma2000. In *Wireless Telecommunications Symposium, 2004*, pages 15–24. IEEE, 2004.
- [9] Abhinav Pathak, Y Charlie Hu, and Ming Zhang. Where is the energy spent inside my app?: fine grained energy accounting on smartphones with eprof. In *Proceedings of the 7th ACM european conference on Computer Systems*, pages 29–42. ACM, 2012.
- [10] Abhinav Pathak, Y Charlie Hu, Ming Zhang, Paramvir Bahl, and Yi-Min Wang. Fine-grained power modeling for smartphones using system call tracing. In *Proceedings of the sixth conference on Computer systems*, pages 153–168. ACM, 2011.
- [11] Arjun Roy, Stephen M Rumble, Ryan Stutsman, Philip Levis, David Mazières, and Nickolai Zeldovich. Energy management in mobile devices with the cinder operating system. In *Proceedings of the sixth conference on Computer systems*, pages 139–152. ACM, 2011.
- [12] Jui-Hung Yeh, Jyh-Cheng Chen, and Chi-Chen Lee. Comparative analysis of energy-saving techniques in 3gpp and 3gpp2 systems. *IEEE transactions on vehicular technology*, 58(1):432–448, 2009.
- [13] Chanmin Yoon, Dongwon Kim, Wonwoo Jung, Chulkoo Kang, and Hojung Cha. Appscope: Application energy metering framework for android smartphone using kernel activity monitoring. In *USENIX Annual Technical Conference*, volume 12, pages 1–14, 2012.
- [14] Heng Zeng, Carla S Ellis, Alvin R Lebeck, and Amin Vahdat. Ecosystem: Managing energy as a first class operating system resource. In *ACM Sigplan Notices*, volume 37, pages 123–132. ACM, 2002.
- [15] Heng Zeng, Carla Schlatter Ellis, Alvin R Lebeck, and Amin Vahdat. Currentcy: A unifying abstraction for expressing energy management policies. In *USENIX Annual Technical Conference, General Track*, pages 43–56, 2003.