Measurement-based Methodology for Modeling the Energy Consumption of Mobile Devices

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

Measurement-based Methodology for Modeling the Energy Consumption of Mobile Devices

Khalil Ibrahim HAMZAOUI

Univ. Lille, CNRS, Centrale Lille,
UMR 9189 – CRIStAL – Centre
de Recherche en Informatique,
Signal et Automatique de Lille,
F-59000, Lille, France

Univ. Mohammed Premier,
FSO, Lab. LANOL
60000, Oujda, Morocco
E-mail: hamzaoui.khalil@gmail.com

Mohammed BERRAJAA

Univ. Mohammed Premier,
FSO, Lab. LANOL
60000, Oujda, Morocco
E-mail: berrajaamo@yahoo.fr

Mostafa AZIZI

Univ. Mohammed Premier,
ESTO, Lab. MATSI
60000, Oujda, Morocco
E-mail: azizi.mos@ump.ma

Giuseppe LIPARI

Univ. Lille, CNRS, Centrale Lille,
UMR 9189 – CRIStAL – Centre
de Recherche en Informatique,
Signal et Automatique de Lille,
F-59000, Lille, France
E-mail: giuseppe.lipari@univ-lille.fr

Pierre BOULET

Univ. Lille, CNRS, Centrale Lille,
UMR 9189 – CRIStAL – Centre
de Recherche en Informatique,
Signal et Automatique de Lille,
F-59000, Lille, France
E-mail: 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, and energy is considered a critical resource, so it is important to know and understand both how energy is measured and consumed on mobile devices. An accurate knowledge will allow us to develop efficient 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 mobile applications. Based on precise measurements, we elaborate predictive models of energy consumption for both unconnected and connected applications.

Keywords: Mobile computing; Operating system; Energy consumption modeling.

1 Introduction

The energy consumption of a smartphone is the amount of energy used to operate its services [8, 15] by the software running on the hardware. Thus, the hardware components used by the applications determine 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 main research presented in this paper consists of developing and practicing a new experimental methodology for modeling the energy consumption in mobile environments and more specifically on the Android platform. The 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 and we describe a methodology to identify the parameters of this model (processor frequency, dissipated power and initial battery level). To this end, we analyzed a collection of experimental data collected during a 33 day long "Tour de France" in a wheelchair [7]. The objectives of our work is to propose a methodology that a developer can use to understand the energy consumption behavior of a device in a specific applicative context. This methodology is based on measurements done on the device under study and leads to the proposal of a mathematical model with identified parameters of the energy consumption of the device. The resulting model can then be used to make decisions to balance energy efficiency and quality of service.

We present the methodology in Section 2, and we validate this methodology on experimental data in Section 3. Then we present related works in Section 4. We conclude our paper in Section 5.

2 Methodology

The research presented here proposes a methodology to model and evaluate the energy costs in mobile environments. The proposed methodology is developed in four stages:

- Data collection;
- Data preparation;
- Modeling for fixed frequencies;
- Modeling for variable frequency.

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 cycle.

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

2.1 Data collection

Scientific data was collected on the basis of several scenarios in different places. These measures were operated 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:

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

Cronoid is an automation tool that allows performing tasks on a regular basis (like cron). It also enables automatic task running when the status of the terminal changes. The version used for this work is Cronoid-3.5.1.

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

The data collection stage of 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 with the CPU Frequency tool.
- The role of the Cronoid tool is to automate tasks in order to minimize interaction with the user.

2.2 Data preparation

After each experiment, the data obtained is stored in a file. Given the large number of parameters and data retrieved, the measurements are organized into two tables. Two examples of such tables are shown in Table 1 and Table 2.

For the implementation of our methodology, we selected the following parameters, obtained by Trepn Profiler:

- 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 the type of correlation between the number of total operations Op and 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 clock cycles of each processor per measurement interval. This active clock cycle 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.

2.3 Modeling

The third 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 the CPU Frequency tool) frequencies. More details on the modeling stages will be given in Section 3 on several case studies. In the first 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 as a function of the frequency F and of Op, the number of operations during this measurement interval. In the case of offline mode study, a linear regression is enough to build a good model. The connected mode case is better managed by a polynomial regression.

In the case of variable frequencies, 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 changes can simply be discarded.

3 Experiments

3.1 Experimental context

Each service activated in a smartphone induces an energy consumption relative to an amount of dissipated energy. The purpose of this study is to provide a detailed monitoring of the main sources of energy consumption. In the first case study, we will use the GSM network, which will be the only one to be in active state (Only the GSM network will be active but unused by the application under test). The other communication networks (3G, 4G, Wi-Fi and the GPS) remain inactive throughout the various experiments. The second case study aims at evaluating and measuring the energy consumed in connected use cases, sources related to the data communications are activated completely (3G / 4G or Wi-Fi) or partially (GPS) depending on the scenarios of the experiment.

Each experiment was carried out over a period of 45 minutes with a rate of ten measurements per second, leading to around 27000 measurements per test case. For the Local Video with Fixed Frequency (LVFF) scenario, twenty experiments were carried out with different frequencies, initial battery levels and locations totaling 540000 measurements. We believe that 10 measures per second is a good compromise between the ability to capture DVFS induced frequency changes to get precise measures and the overhead induced by the measuring process. Indeed there are on the order of 100,000 processor cycles per measuring interval (at 1 GHz).

To collect the data, a tour of France was carried out using an electrical wheelchair to perform measurements in a diversified environment based on several scenarios. Figure 1 details the route of the measurement sites (33 days and 3006 km).

All measurements were performed on an HP Pro Slate 8 tablet that has the characteristics listed in Table 3.

3.2 Modeling of energy consumption

In this section, we present two cases. The first case corresponds to the disconnected mode, where we study local video playing with a fixed frequency while driving to test the impact of the variation of the GSM network on the energy consumption in disconnected mode.

3.2.1 Modeling in offline mode

The disconnected mode disables all data connection components such as Wi-Fi, Bluetooth and GPS. In this mode, we used local data for energy modeling.

First scenario, Local Video with Fixed Frequency (LVFF): 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 a fixed in advance frequency.

Second scenario, Local Video with Variable Frequency (LVVF): 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.

3.2.2 Modeling in connected mode

The second case corresponds to the connected mode. On the one hand, we study the case of a remote video with the fixed and variable frequency in motion to test the impact of the variation of the GSM network 3G / 4G or Wi-Fi on the energy consumption in connected mode. On the other hand we study the case of navigation where, in addition to activating the GSM network 3G / 4G or Wi-Fi and Bluetooth, we added the GPS for testing the impact of navigation on the energy consumption.

The active applications during the measurements are: Facebook, GsmaService, 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 the video scenario, we worked 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 initial battery levels.

The aim of this part is to evaluate the dissipated energy per clock cycle which will be called "dissipated elementary energy".

To qualify the quality of the model, we systematically show 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 cycle can be expressed as follows:

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

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

3.3 Results of measurements

3.3.1 Disconnected mode case

The various measures relating to the energy dissipated were carried out in relation to the number of total operations at a fixed frequency. Figure 2 shows an example of experimental data obtained 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 in Figure 3.

The respective equations of the regressions obtained are:

$$E(1.0, Op) = 0.0512 + 5.995.e^{-11}Op, R^2 = 0.830$$
$$E(1.5, Op) = 0.0321 + 4.226.e^{-11}Op, R^2 = 1.000$$
$$E(1.7, Op) = 0.0337 + 5.073.e^{-11}Op, R^2 = 1.000$$
$$E(2.2, Op) = 0.0309 + 7.210.e^{-11}Op, R^2 = 1.000$$

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 the degradation of the quality of the video (following the frequency imposed in the experimentation) which explains the obtained result. The low R^2 is due to the fact that the required computing power to decode the video is too high for a 1 GHz frequency, thus the QoS is degraded and it shows on the R^2 coefficient.

On the basis of the experimental results obtained, we can deduce the energy discharge per clock 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 4.

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 5.

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

$$E(F, Op) = E_{\text{dynamic}}(F) \times Op + E_{\text{static}}(F)$$

Where E is the total energy, F the selected frequency, Op is the number of operations per measurement interval (in our case 100 ms).

In this study, the coefficients related to the LVFF scenario are thus defined by the following equation:

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

3.4 Validation of the mathematical model

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

The experiment that is used for the test consists 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 coherency with a relatively low error rate.

The average relative error between our model and the experimental measurements 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 of the difference between the measurements obtained experimentally and the values predicted by our model 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 Figure 6. 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.

3.5 Connected mode case

The purpose of this Section is to study energy consumption while activating Wi-Fi. To stay in the same context, the remote video was identical to the one used in disconnected mode. We determine the type of correlation between the energy consumption and total operations, and we crosscheck with the measurements obtained in the offline mode.

Figure 7 shows an example of experimentation by fixing the frequency of the processor at 1.5 GHz for a remote video with fixed frequency (RVFF).

Figures 8 and 9 show respectively that the static energy and dynamic energy of the RVFF model vary linearly with respect to the frequency.

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

$$E_{\text{Dynamic}}(F) = 5.002.e^{-11} \times F - 2.104.e^{-11}, R^2 = 0.958$$
$$E_{\text{Static}}(F) = 0.03 \times F + 0,03, R^2 = 0.780$$

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

$$E(F, Op) = (5.002.e^{-11} \times F - 2.104.e^{-11}) \times Op + (0.03 \times F + 0,03)$$

3.6 Comparative study between LVFF and RVFF scenarios

In this part we discuss the energy gap existing between the LVFF scenario and the RVFF scenario under similar conditions (with the same frequency). To compare the energy variation between the two scenarios, we compute the difference between the measurements obtained experimentally with the RVFF scenario and consumptions predicted by the LVFF model. This difference corresponds the communication cost and is shown in Figure 10.

The energy consumption related to the communication is well modeled according to a quadratic regression which gives a better result, more relevance and more simplicity than a linear regression.

3.6.1 Navigation case

In the navigation case we activate both data connection and the GPS to monitor the energy consumption behavior.

Table 5 shows that the navigation uses a lot of CPU resources (19.14 %) and thus dominates the total energy consumption of the system.

The first case is to set the frequency and to track the total dissipated energy of the active applications separately at first, then the overall consumption of the Navigation with Fixed Frequency (NFF) scenario. Figure 11 shows an example of experimentation relating to the NFF scenario along with the regression. Figures 12 and 13 show the linear regressions leading to the variable frequency NFF model.

3.6.2 Navigation with variable frequency

This part deals with the case of navigation with variable frequency (NVF), the frequency being assigned by the operating system. Figure 14 presents the results of NVF experiments. We note the existence of a data flow that strongly depends on the frequency at each measurement. The large number of operations and the rate of charge explain the shape of the obtained graph.

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

$$E_{\text{Dynamic}}(F) = 4.002.e^{-11} \times F - 2.103.e^{-11}, R^2 = 0.999$$
$$E_{\text{Static}}(F) = 0.098 \times F - 0,0391, R^2 = 0.940$$

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

$$E(F, Op) = (4.002.e^{-11} \times F - 2.103.e^{-11}) \times Op + (0.098 \times F + 0,0391)$$

3.6.3 Comparative study between NFF and NVF scenarios

Here we discuss the energy gap between the NFF and NVF scenarios. To compare the energy variation between the two scenarios, we compute the difference between the measurements obtained experimentally with the NVF scenario and the values predicted by the NFF model to determine the cost due to the variable frequency. This difference is shown in Figure 15.

The result obtained mathematically based on the difference on the respective equations of the NVF and NFF scenarios is very close to that obtained experimentally, the results obtained are as follows:

$$NFF : 0.05 + 5.e^{-11}x \tag{1}$$

$$NFV : 0.07 + 2.e^{-11}x + 5.e^{-21}x^2 \tag{2}$$

$$NFV - NFF : 0.02 - 3.e^{-11}x + 5.e^{-21}x^2 \tag{3}$$

The static energy remains constant in the different experiments, the dynamic energy difference measured experimentally for the NFF and NFV scenarios was compared with the energy obtained by the proposed model.

The equation obtained by the model is expressed as:

$$0.03 + 2.e^{-11}x + 2.e^{-21}x^2 \tag{4}$$

The value of the dynamic energy obtained by the model (Equation 4) is close to that obtained experimentally (Equation 3).

The difference obtained is due to the variability of the frequency (dynamic energy) and presence of tail energy which is relative to the devices (Wi-Fi, GPS ...) which are not necessarily permanently active during the experiments.

In this study, we treated the different types of dependencies between energy dissipation and frequency of multiple scenarios in offline mode (LVFF) and the connected mode (RVFF, NFF, NVF). The difference between the scenarios can be explained on the one hand by the variability of the received signal strength for GPS and Wi-Fi (for connected mode) which forces the activation / deactivation of the services that will generate an additional consumption due to the tail energy. On the other hand the scenarios based on the connected mode require the activation of additional devices such as the GPS, which increases the rate of the inputs / outputs which are partially responsible for additional energy cost.

4 Comparison with the state of the art

Several research studies have also proposed CPU frequency-scaling techniques by taking into account the quality of user experience [10, 8].

4.1 Related Modeling Works

4.1.1 TailEnder

Balasubramanian, Balasubramanian and Venkataramani designed TailEnder [1] for measuring the characteristics of the energy consumption of 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 has important and non-intuitive implications for the design of efficient energy application. Previous studies such as [9, 16, 2] have studied the impact of different energy-saving techniques in 3G networks using analytical models [8].

4.1.2 Cinder

Roy, Rumble, Stutsman, Levis, Mazieres and Zeldovich show in [14] 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 [14] consists of detecting the main energy-consuming tasks and then assigning this information to the guiding applications in order to allocate the energy resources to the highest priority tasks [18, 19, 4, 3].

4.1.3 BLAST

Mercati, Hanumaiah and Bloch have shown in BLAST (Battery Lifetime-constrained Adaptation with Selected Target) [11] that performance can be maximized if the user can select a target battery lifetime.

This framework is an application-aware power management framework for mobile devices which controls operating conditions in order to meet a predefined battery lifetime.

4.1.4 Eprof

Pathak, Hu and Zhang have shown in Eprof [12] that some of the components such as Wi-Fi, 3G GPS have a tail behavior [13] in which a component can enter a state of high power and remain in that state of energy beyond the end of the routine.

This approach evaluates the energy consumed by the I/O and the energy of the tail [13, 1] as well as the energy consumed because of wake-up caused by the wake locks.

4.1.5 RAPL

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

RAPL is based on energy sensors available for measuring the energy consumption, and takes into account the energy consumption in the software components.

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

4.1.6 AppScope

AppScope is a system that automatically evaluates the energy consumption of applications running on

Android smartphones [17]. Its design is based on monitoring the Android kernel at a microscopic level. The objective of the AppScope model is to measure the energy consumption of applications using power equipment models and usage statistics for each hardware component.

4.2 Synthesis

We have previously published a more detailed study on the state of art in [8].

Table 5 compare these energy consumption methodologies and the one we propose in this article on different criteria. Here are the meanings of the columns of the Table:

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

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

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

QoS impact on the quality of service (QoS), 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 malwares to reduce battery life.

Dev developer help, this criterion is used to reduce the energy consumption of a software during its design.

Eprof and our model HBALB (HBALB corresponds to the initial letters of the names of the authors of this work) 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. HBALB allows to model the energy consumption so that it is optimized for better performances.

TailEnder, Cinder, RAPL satisfy the “Hard opti.” criterion. TailEnder performs a minimization of the energy consumption for the application which tolerate a small delay by quantifying the use of the energy, and 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, BLAST and HBALB enable 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. BLAST maximize performance while letting the device battery to last at least for a certain required lifetime. Our model HBALB allows to define the optimal parameters for a better performance which justifies their relevance in the “Soft optim.” criterion.

The quality of service is well managed by Cinder because it automatically manages the allocation of energy resources, the HBALB 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 prioritizes 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 by analyzing the different energy states of the device (characteristics, tail energy etc).

Our proposal provides precise models with respect to the active applications, which is not treated by the other case studies.

4.3 Advantages of the proposed methodology

- Our methodology 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.

4.4 Limits of the proposed methodology

- The precision of the models 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.
- The methodology currently lacks GPU and modem energy consumption modeling. It relates the global energy consumption of the whole device to only the activity of the CPU, and so lacks precision when the energy consumption is heavily influenced by other components.

5 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 as a function of the activity of the processing cores for fixed frequencies by (linear) regression. The last step in our methodology consists in elaborating a model of the energy consumption as a function of the operating frequency and the activity of the processor by a regression on the parameters of the fixed frequency models.

Experiments related to this methodology have been carried out on different scenarios:

- Local video case;
- Remote video;
- Navigation.

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.

For connected use cases, we notice that the prediction is less precise given the number of parameters involved and the diversity of situations encountered. It seems necessary in these cases to take into account more precisely the communication costs. The question on how to do that is left for future work.

The main quality of our methodology comparing with 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, it would be useful to extend this methodology to deal with other use cases that exercise other components of the complex mobile hardware platform. Checking the contribution of the GPU to the overall energy consumption for gaming use cases would be an interesting study. 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, Mohammed Berrajaa, Mostafa Azizi, Giuseppe Lipari, and Pierre Boulet. An experimental methodology for modeling the energy consumption of mobile devices. In *Embedded & Distributed Systems (EDiS), 2017 First International Conference on*, pages 1–6. IEEE, 2017.

[8] 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.

[9] 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.

[10] Xueliang Li, Guihai Yan, Yinhe Han, and Xiaowei Li. Smartcap: user experience-oriented power adaptation for smartphone's application processor. In *Proceedings of the Conference on Design, Automation and Test in Europe*, pages 57–60. EDA Consortium, 2013.

[11] Pietro Mercati, Vinay Hanumaiah, Jitendra Kulkarni, Simon Bloch, and Tajana Rosing. Blast: Battery lifetime-constrained adaptation with selected target in mobile devices. In *proceedings of the 12th EAI International Conference on*

Mobile and Ubiquitous Systems: Computing, Networking and Services on 12th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, pages 80–89. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2015.

[12] 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.

[13] 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.

[14] 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.

[15] Wook Song, Nosub Sung, Byung-Gon Chun, and Jihong Kim. Reducing energy consumption of smartphones using user-perceived response time analysis. In *Proceedings of the 15th Workshop on Mobile Computing Systems and Applications*, page 20. ACM, 2014.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

Table 1 Sample of measured data (part 1)

Time (ms)	Cpu1 Freq (kHz)	Cpu1 Load %	Cpu2 Freq (kHz)	Cpu2 Load %	Cpu3 Freq (kHz)	Cpu3 Load %	Cpu4 Freq (kHz)	Cpu4 Load %
0	652800	85	729600	80	729600	60	729600	71
100	652800	88	729600	82	729600	66	729600	80
200	652800	73	729600	81	729600	83	729600	60
300	652800	100	729600	82	729600	75	729600	75
400	652800	80	729600	83	729600	75	729600	83
500	652800	83	729600	88	729600	83	729600	80
600	652800	92	729600	86	729600	100	729600	80
700	729600	87	729600	85	729600	100	729600	87
800	729600	66	729600	90	729600	87	729600	75
900	729600	80	729600	91	729600	71	729600	83
1000	729600	80	729600	82	729600	83	729600	75
...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...
27000	300000	90	883200	90	729600	80	729600	80

Table 2 Sample of measured data (part 2)

Time (ms)	Memory Usage (Kb)	Screen Brightness	Battery Power (uW)	Battery Remaining (%)	GPU Load (%)	CPU Load (%)
0	1800004	255	875708	3665025	45	89
100	1800376	255	866186	3665076	45	86
200	1800516	255	831271	3665121	45	83
300	1800648	255	880999	3665165	45	76
400	1800896	255	868302	3665230	45	76
500	1801152	255	890521	3665275	45	77
600	1801504	255	948712	3665334	45	82
700	1801636	255	920146	3665387	45	65
800	1802140	255	891579	3665439	50	82
900	1802404	255	882057	3665495	50	68
1000	1802660	255	893862	3665440	50	75
...	...	...	...	...	...	...
...	...	...	...	...	...	...
27000	1802410	255	883200	3665404	47	82

Table 3 Characteristics of the experimental equipment

Type of Product	**Tablet**
Operating system	Android 4.4.4 (KitKat)
Processor	QUALCOMM Snapdragon 800
Max. CPU frequency	2.3 GHz
Number of cores	4
Sensors	Accelerometer, ambient light, proximity, compass, barometer, gyroscope, Hall effect
Number of batteries	1
Technology	Lithium polymer
Autonomy	Up to 13.75 h
Operating temp mini	0°C
Operating temp maxi	40°C
Storage temp mini	-20°C
Storage temp maxi	60°C

Table 4 Distribution of system resources for the NFF scenario

Applications	**CPU %**
Maps	**19.14**
Parameters	3.93
Messenger	0.75
Interface	0.45
Power Battery	0.4
Cronoid	0.38
Configuration of	0.24
Google Play Music	0.23
Facebook	0.09
Gmail	0.072
Cron Tasker Free	0.03
Hangouts	0.022
Drive	0.005
Application Launcher	1.48E-4
CPU Frequency	5.51E-5
Firefox	5.38E-5

Table 5 Comparison of energy consumption methodologies

	Model quality	**Hard optim.**	**Soft optim.**	**QoS**	**Security**	**Dev.**
TailEnder	+	++	NA	+	NA	NA
Cinder	+	++	NA	++	+++	NA
Eprof	++	+	+	+	NA	++
RAPL	+	++	NA	+	+	+
AppScope	+	NA	+++	+	NA	+
BLAST	+	NA	++	+	NA	NA
HBALB	++	+	++	+	**NA**	+

Figure 1 Route of the scientific tour of France

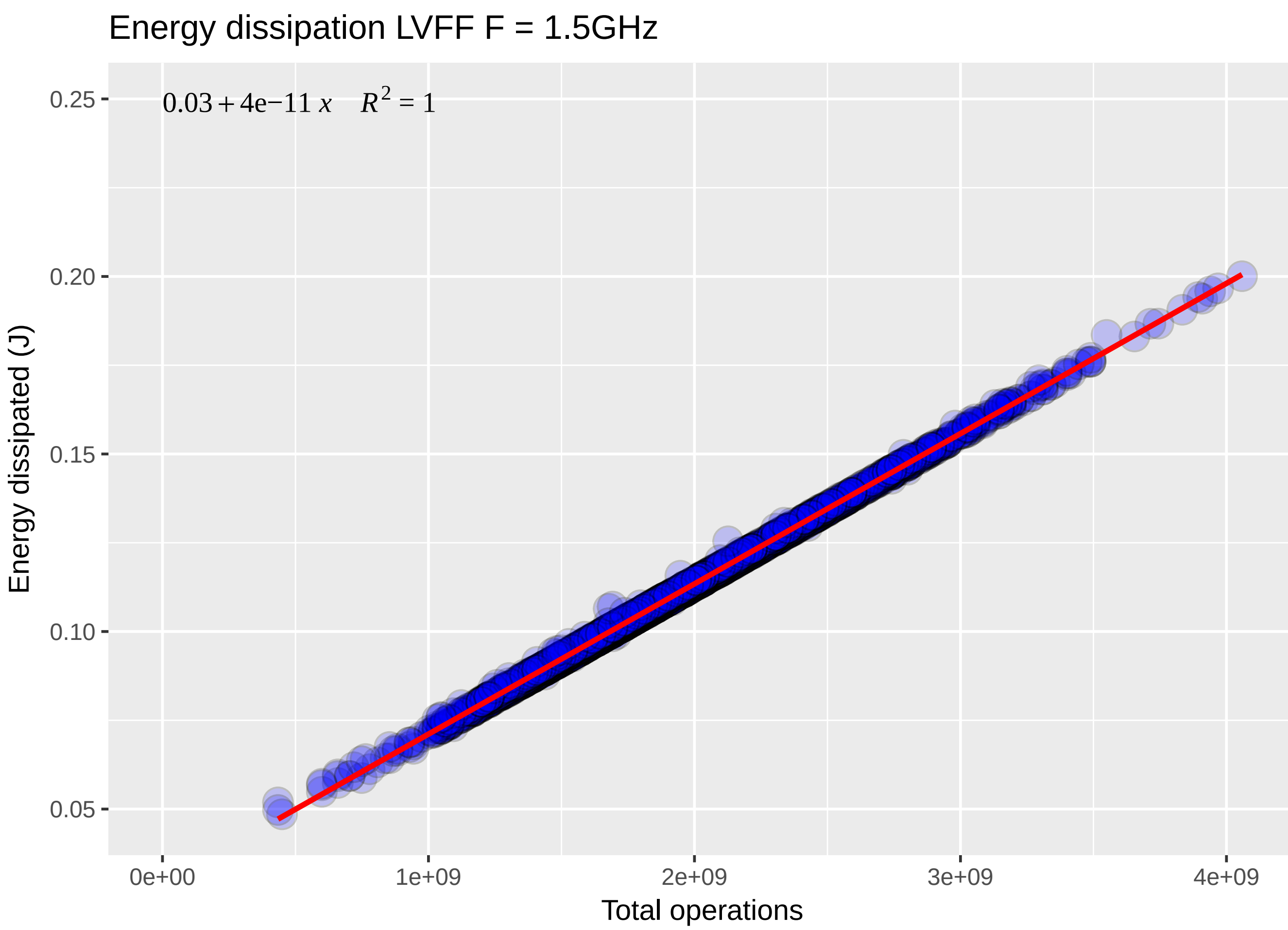

Figure 2 Energy dissipation for LVFF model, Frequency=1.5GHz

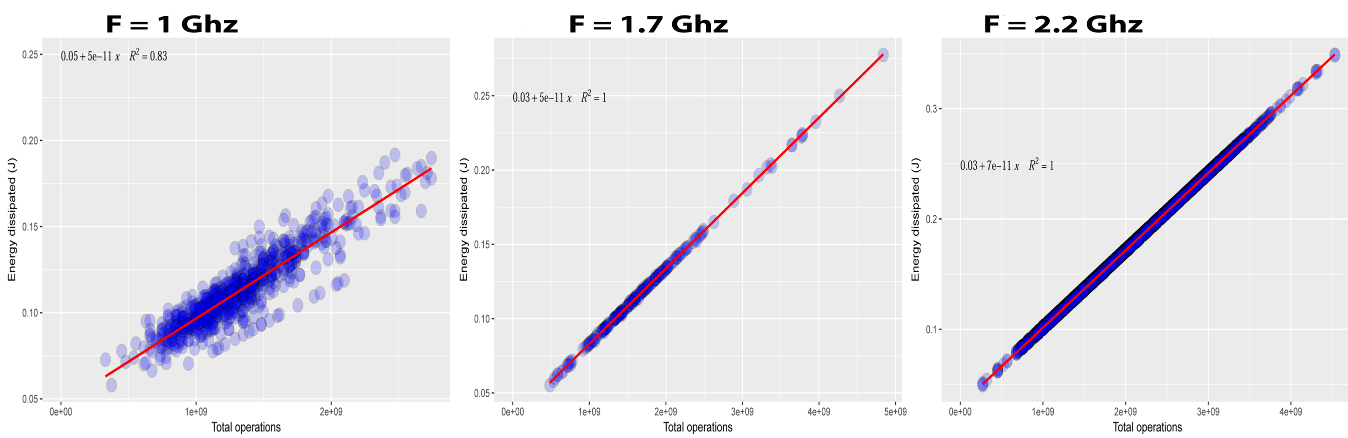

Figure 3 LVFF frequencies = 1 GHz, 1.7 GHz, 2.2 GHz

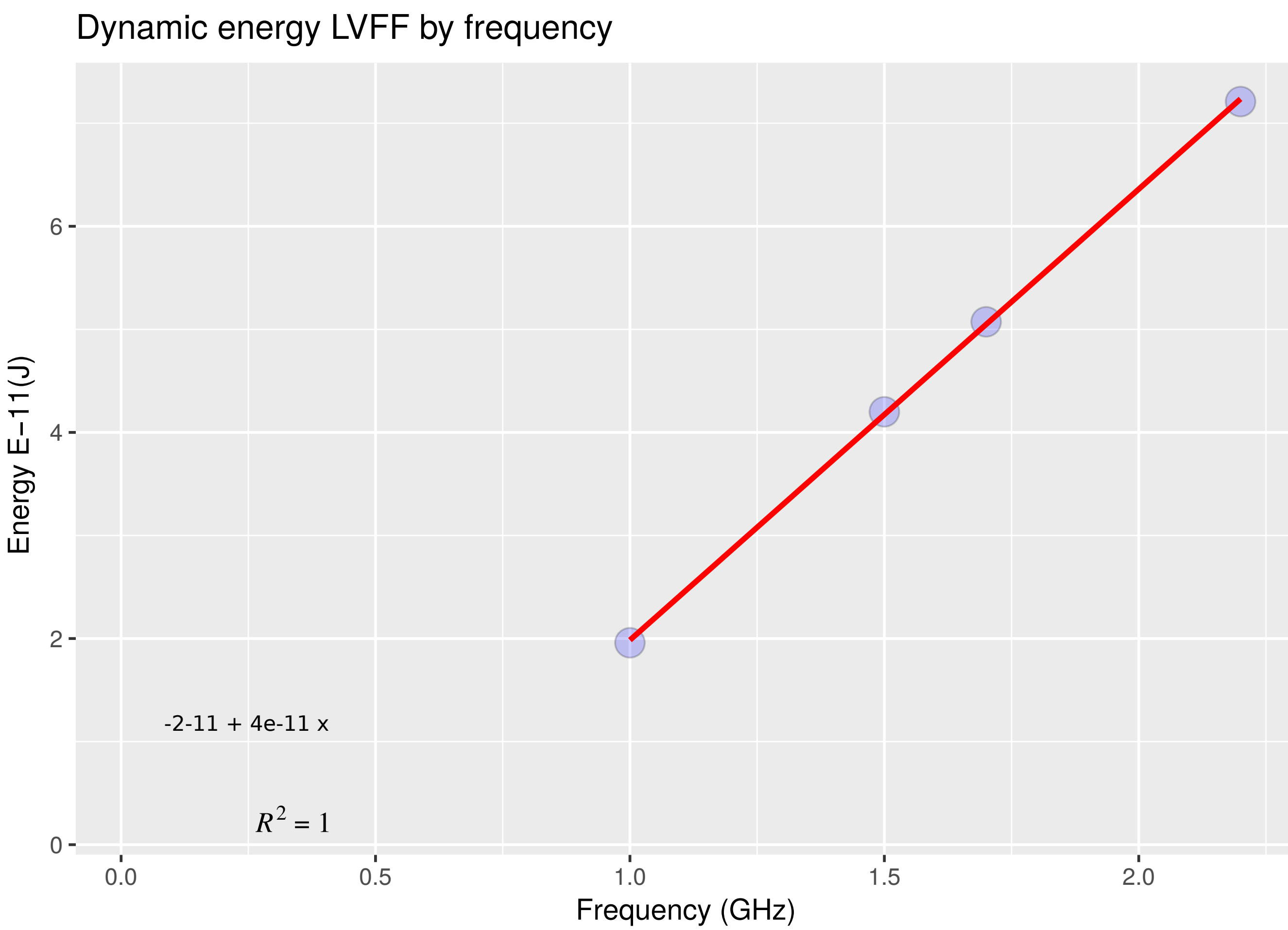

Figure 4 Dynamic energy dissipated by clock cycle in function of the frequency for the LVFF scenario

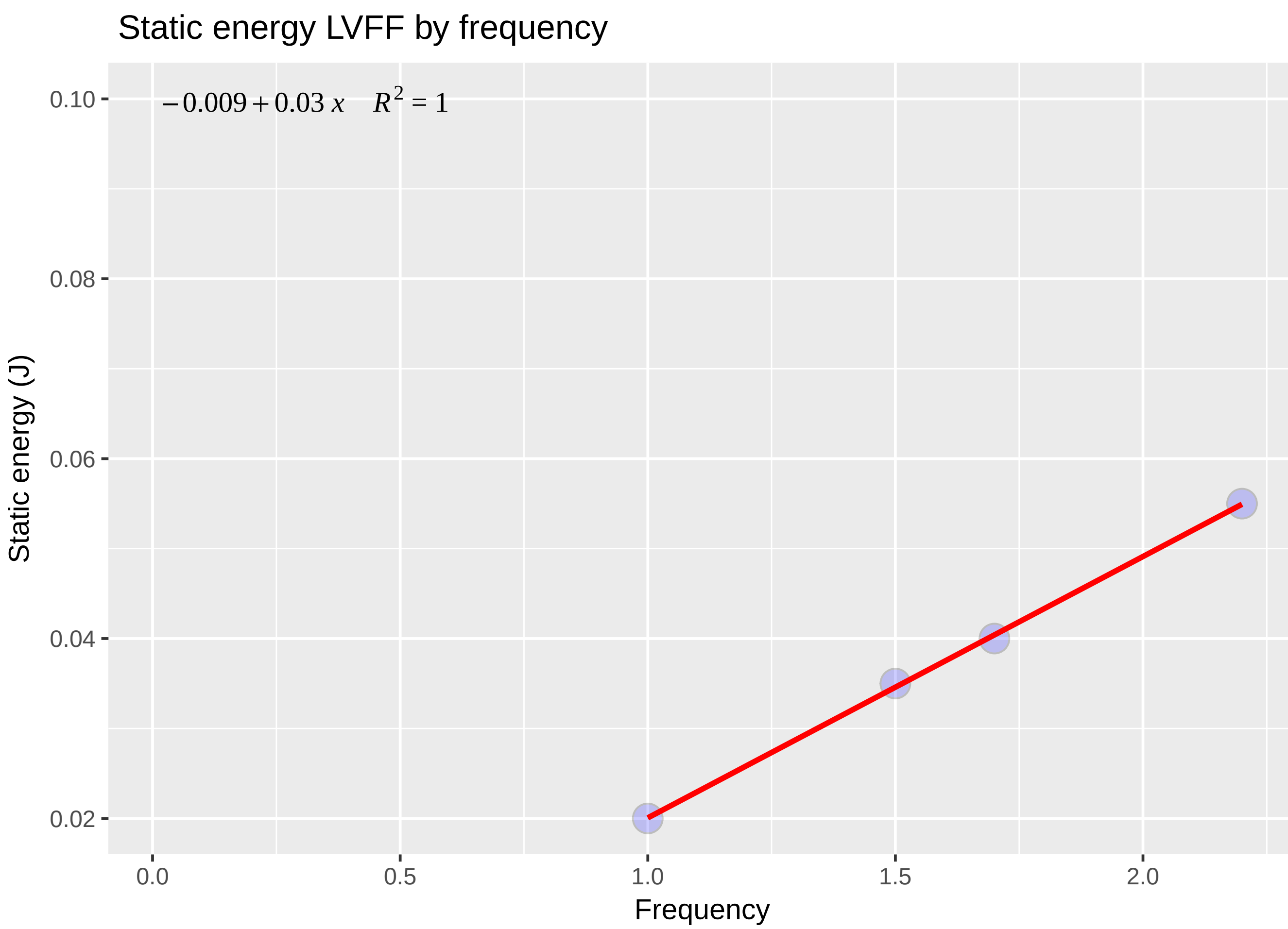

Figure 5 Static energy in function of the frequency for the LVFF scenario

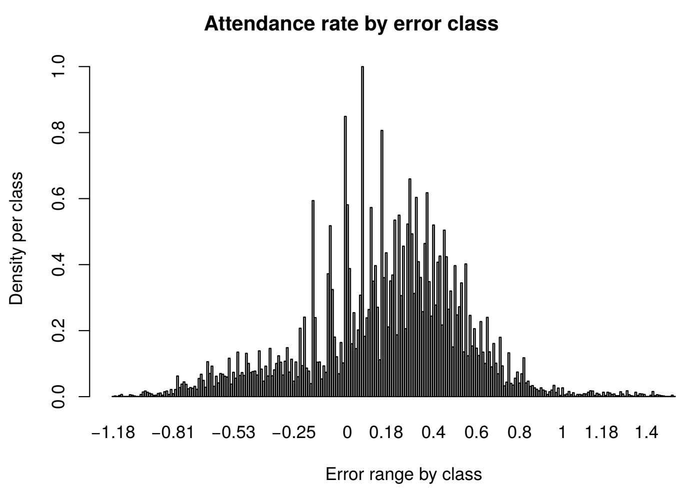

Figure 6 Normalized error distribution using 1000 error class

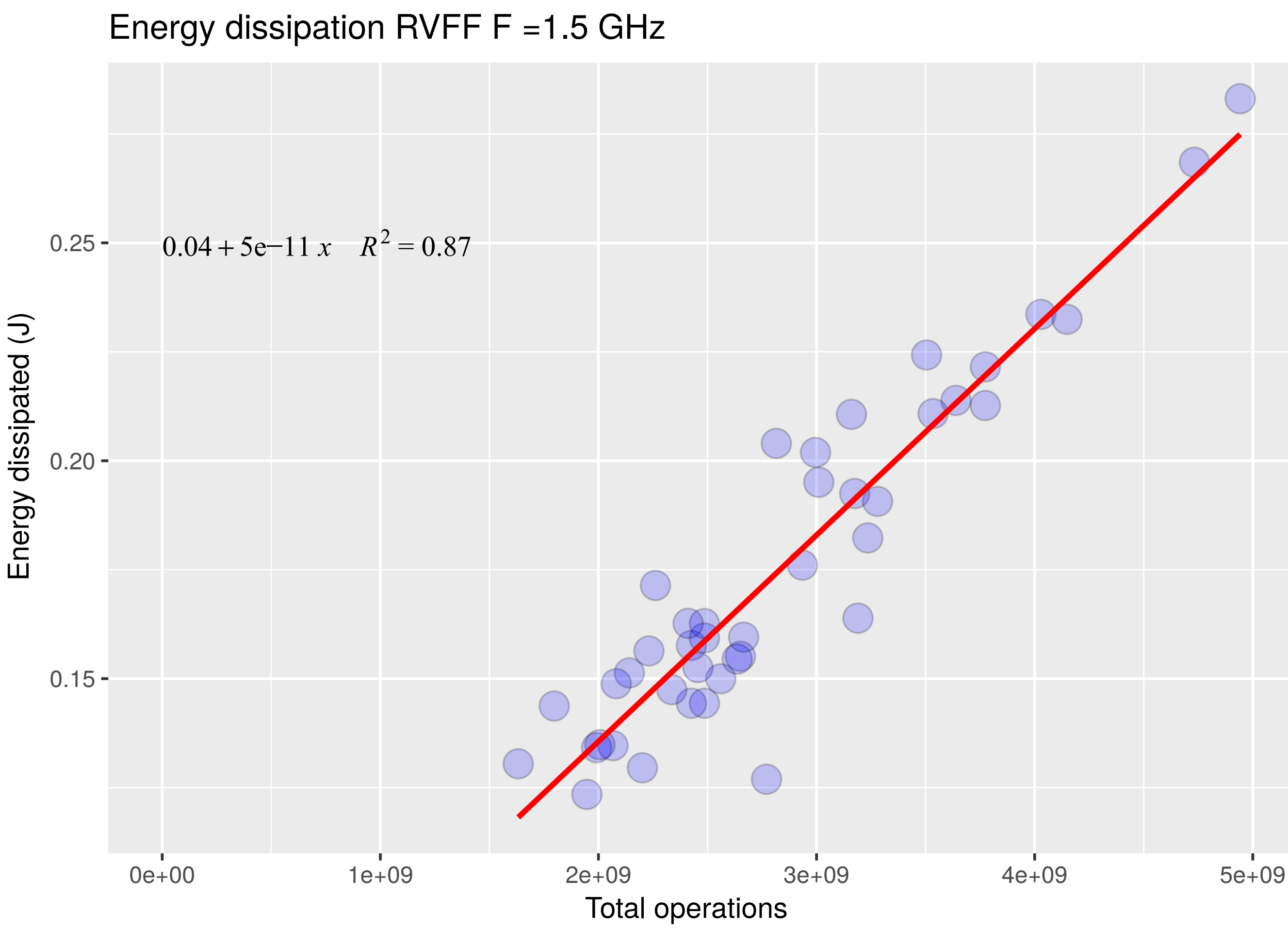

Figure 7 Energy dissipation for RVFF scenario, Frequency=1.5GHz

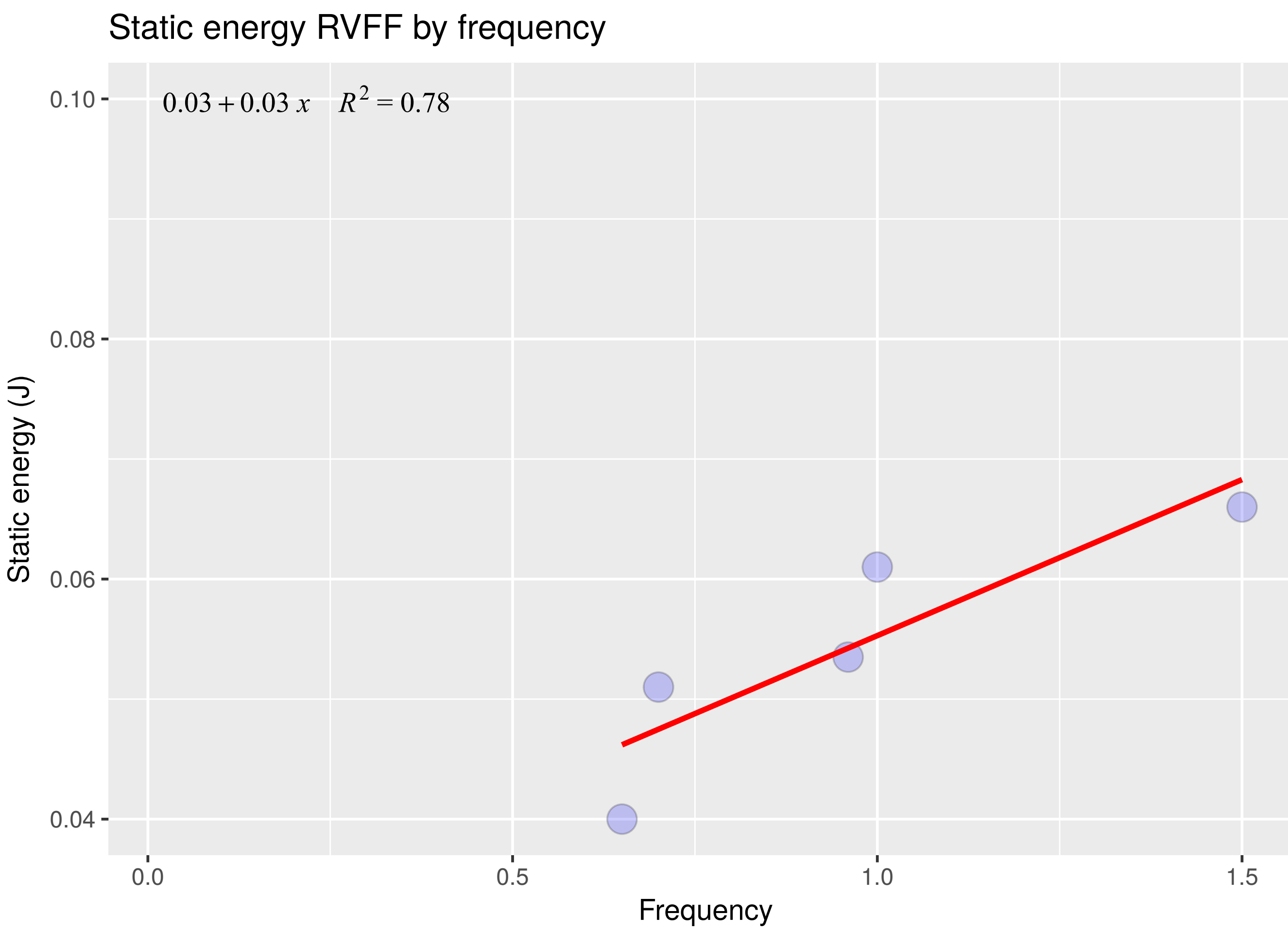

Figure 8 Static energy in function of the frequency for the RVFF scenario

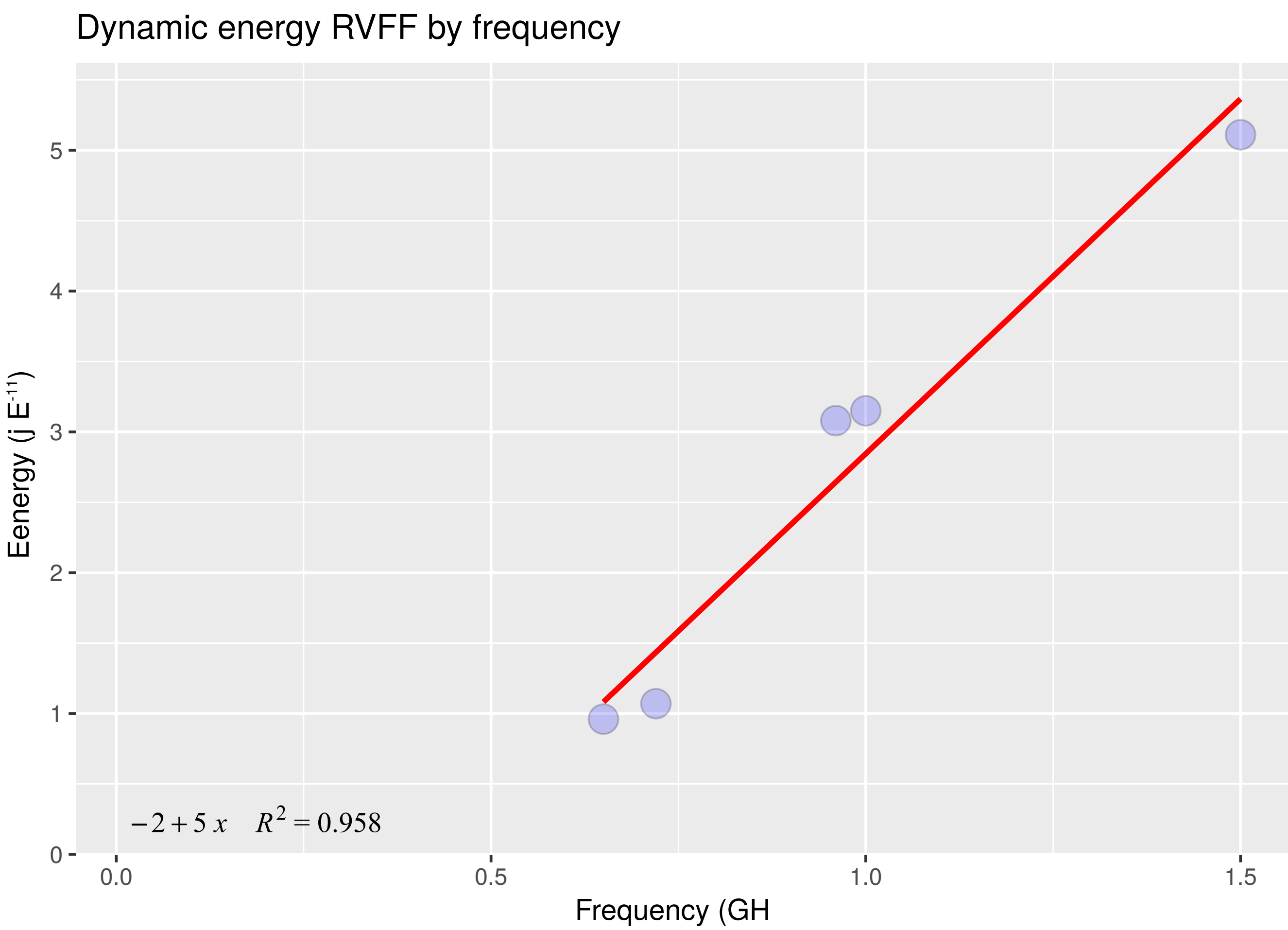

Figure 9 Dynamic energy dissipated by clock cycle in function of the frequency for the RVFF scenario

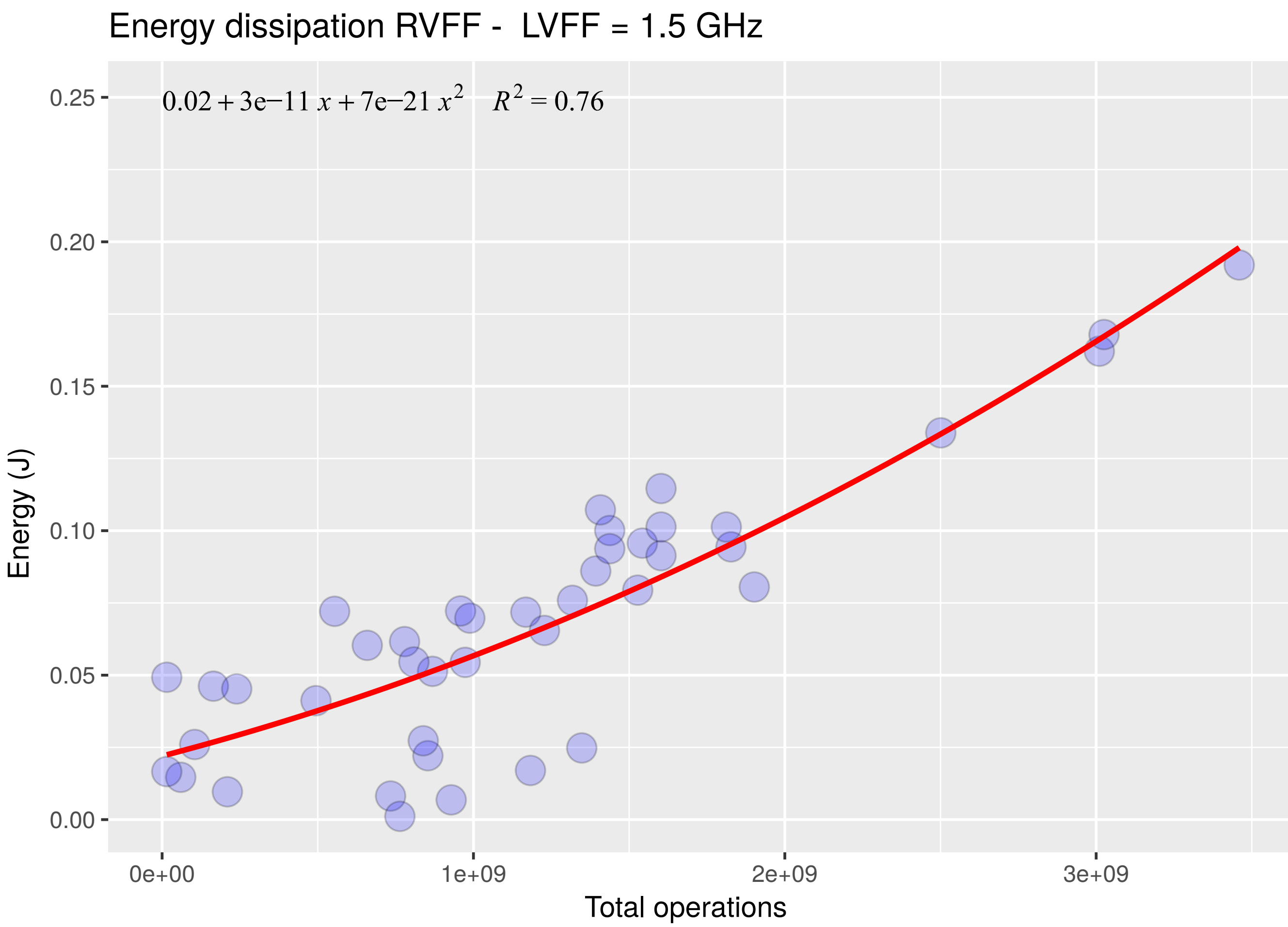

Figure 10 Difference of energy consumption between the measures of the RVFF scenario and the predicted values of the LVFF model.

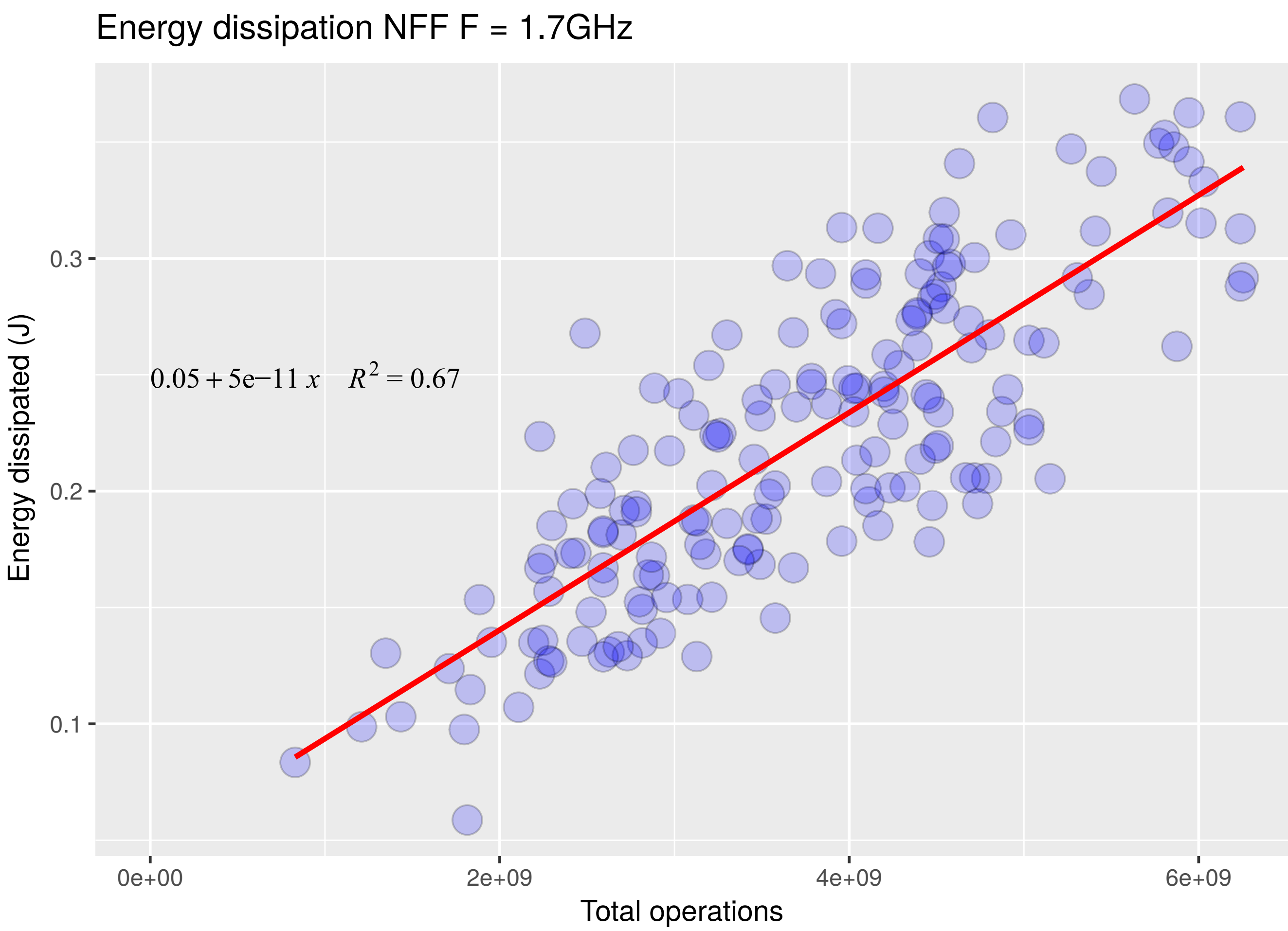

Figure 11 Energy consumption by total operations for the NFF scenario with a fixed 1.7 GHz frequency

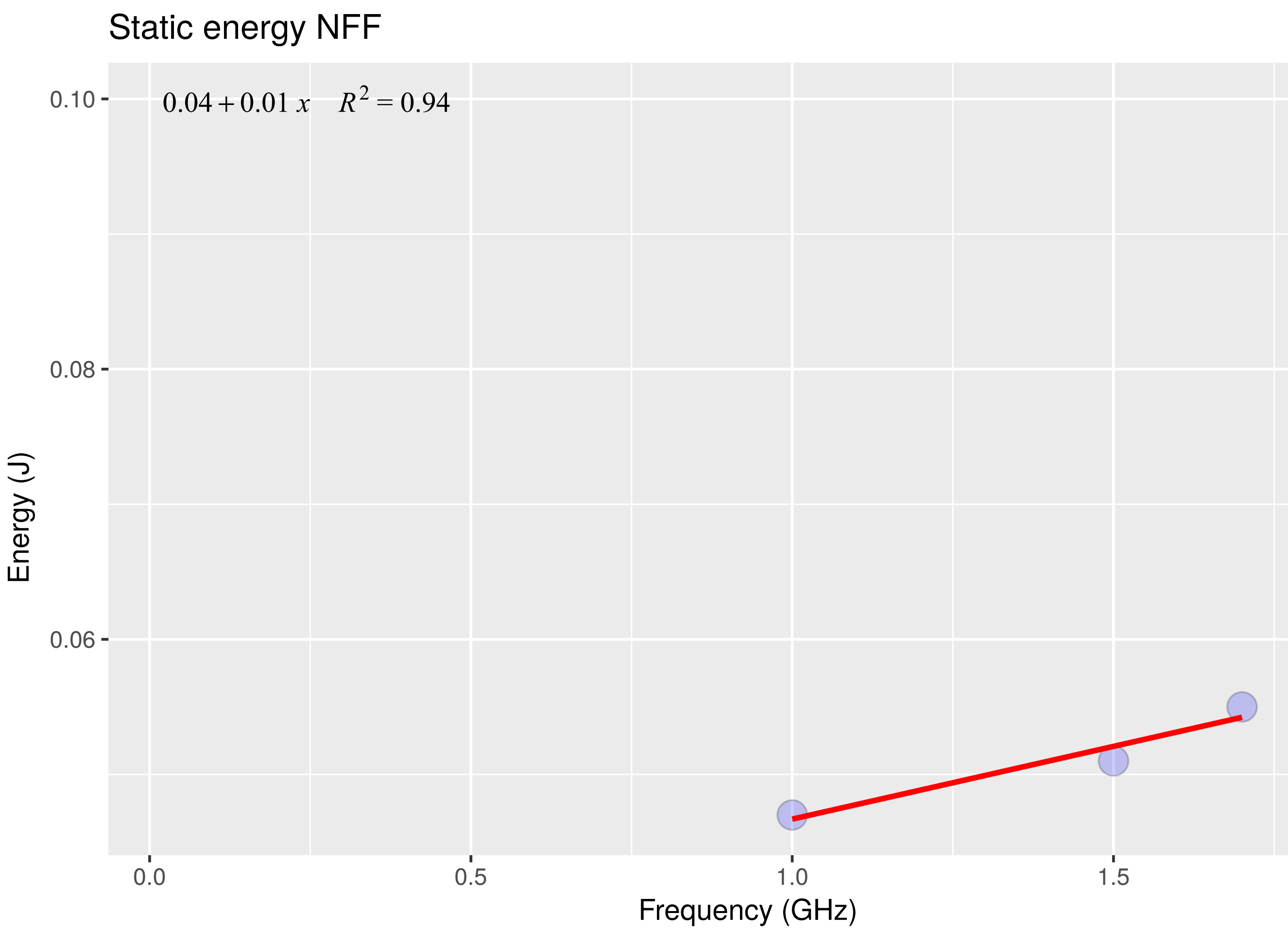

Figure 12 Static energy in function of the frequency for the NFF scenario

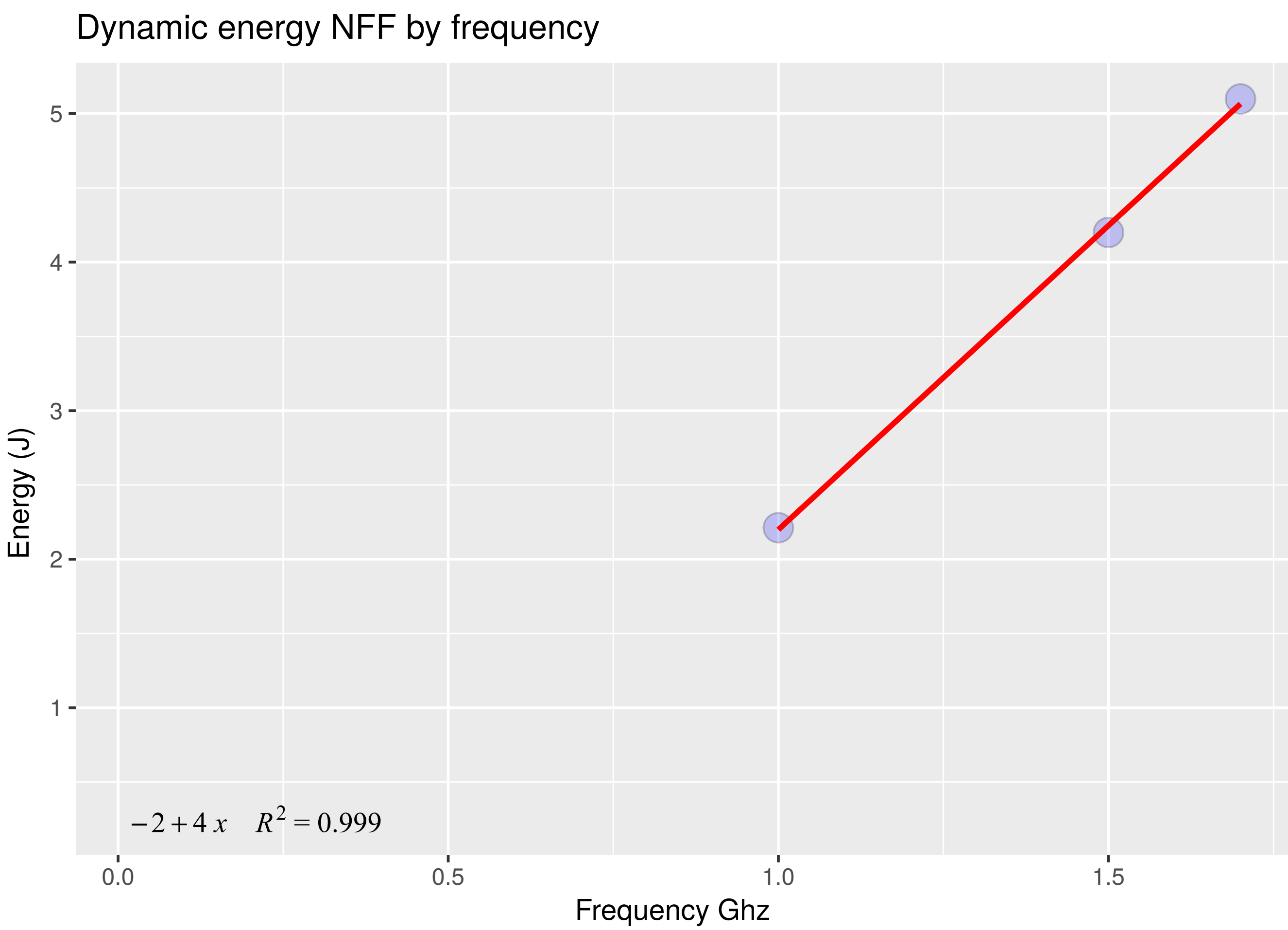

Figure 13 Dynamic energy in function of the frequency for the NFF scenario

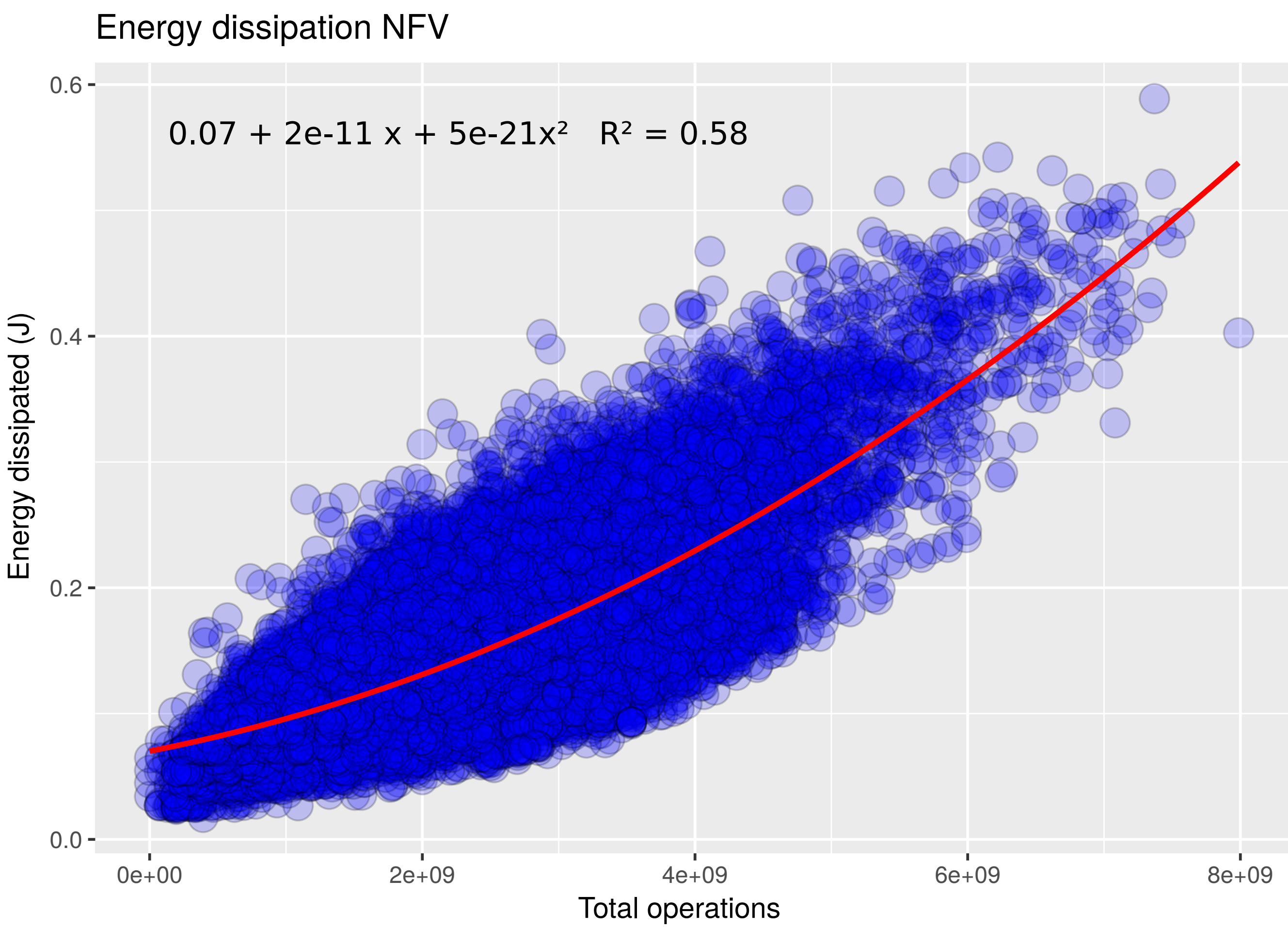

Figure 14 Energy consumption by total operations for the NVF scenario

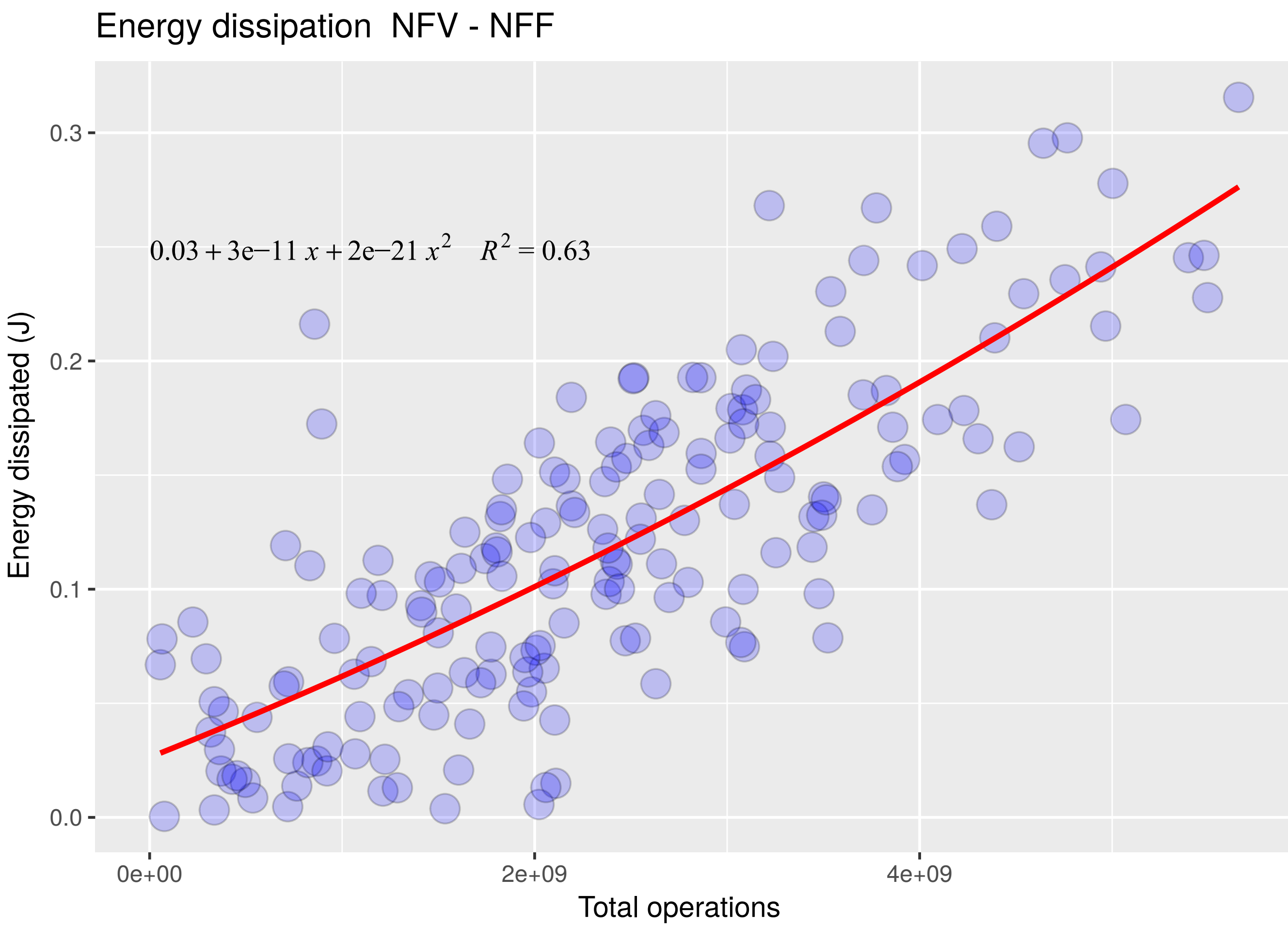

Figure 15 Difference of energy consumption between the measures of the NVF scenario and the predicted values of the NFF model.