Statistical Analysis of Processor Frequency Scaling Algorithms

Paper ID 57
XXX
XXX

Paper ID 57
XXX
XXX

Abstract

Recently can be observed a growth in mobile device sales, including smartphones and tablets, with its high quantity and diversity of hardware and software models, it put on focus the fragmentation problem. Batteries with low capacity are constraints in design of embedded systems. The increase of features highlighted the need for greater capacity in mobile batteries to provide an extended time of use. This work proposes an environment for simultaneously execute tests on mobile devices with support for energy analysis. Two major technology smartphones were used to evaluate the proposed environment. Experiments to provide energy efficiency optimizations in mobile processor frequency scaling were conducted. From the analysis made in the environment, techniques using DVFS that allowed energy savings of up to 25% were created.

Categories and Subject Descriptors CR-number [*subcategory*]: third-level

Keywords Android, Governor, Energy Reduction

1. Introduction

Devices such as smartphones and tablets that use the Android operating system grew surprisingly, increasing from 200.000 activated per day in 2010 to 1.5 million devices activated per day in 2013 and actually is the platform with more active devices [15]. This growth can be explained by the amount of features that are aggregated to smartphones and tablets each year, reducing costs of equipment and the variety of options that encompass the various user profiles, from simple and inexpensive to more complex solutions.

In the world the most used operating system by these devices is the Android. It is built based on the contributions of the open-source Linux community, owning more than 300 versions of hardware and software. The open source Android code and the easy way to publish applications in the virtual store favored to make it the favorite for consumers and developers. Currently are more than 1.5 billion apps on Google Play [5].

The need to prolong the use of devices, the strong growth of mobile devices and Android operating system, the breakthrough in the development of multicore processors, it becomes necessary to develop techniques that reduce the energy consumption of these devices to increase its time of use. This is a concern of most mobile devices design, among these are smartphones and tablets.

Devices with more features require more processing power and consequently more energy dissipation. This processing power is measured by the amount of processors and cores and the clock frequency of each core of the processor. For developing techniques that allow energy efficiency it is necessary to understand the problem addressed and the characterization of their energy spent on mobile. To perform the characterization is necessary to have an energy measurement infrastructure.

The motivation of this work is the lack of a measurement environment (hardware and software integration) that assists in characterizing the energy consumption of mobile devices and allows automate tests. The characterization is a necessary step in the development and testing of new techniques to reduce power consumption in mobile devices.

The remainder of this paper is organized as follows: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

2. Background

Energy measurement is the evaluation of the total energy required to perform an amount of work. Energy is the amount of joules spent in a given time and power is the rate at which energy is expended, the latter is measured in watts (joules / second). To measure how much energy is spent on a device, it must know the expended power in a given time. To calculate the power is necessary to have the current and voltage samples that pass through the device.

To estimate the amount of energy expended in a device is necessary to calculate the area under the graph of instantaneous power by experiment time. The trapezoid method [3] can be used to calculate the chart area that represents the

CONF 'yy, Month d–d, 20yy, City, ST, Country.

http://dx.doi.org/10.1145/nnnnnnn.nnnnnnnReprinted from CONF 'yy, [Unknown Proceedings], Month d–d, 20yy, City, ST, Country, pp. 1–6.

energy consumed by a device. From the current and voltage samples is calculated the instantaneous power which is plotted on a graph over time. The method works basically to slice the chart area in n pieces, where each piece forms a trapeze and so the area can be measured. The integral of each interval is calculated and all intervals are summed to obtain the approximation of the energy amount expended.

2.1 Linux and Android Governors

The Linux operating system uses governors for operational control system. Governors are regulatory modules of the Linux kernel that communicate with the hardware. The main function is to regulate the operating frequencies of the processor cores. The Android OS is based on Linux kernel and also uses governors. There governors that focus on performance, while others, in energy savings. This choice must be made according to user usage profile [16].

Performance and energy spent minimization are opposite variables. When improve one, worse another. It is necessary to balance these variables to provide the user a better user experience. The governors more used in Linux operating system are: OnDemand, Performance and PowerSave. For Android operating system has that ones used in Linux with others adapted for mobile devices, such as Interactive, PegasusQ, UserSpace, among others.

2.2 Techniques to Reduce Energy Consumption

Several techniques that reduce energy consumption in mobile devices are available in literature. You can perform energy optimizations in several areas, such as data transmission: choose intelligently and in runtime which data network is better to use, Wifi or mobile networks (2G / 3G / 4G) [8], decreased downtime at the end of data transmissions, this technique is called fast dormancy [7], grouping data packets and send by bursts (batching) [13] [11], decouple processing the transmission of data [7], processing off-load to the cloud, Dynamic Voltage and Frequency Scaling (DVFS) [2], among others.

In DVFS technique is possible to reduce the operating frequency of the processor or the voltage to decrease the power expended. In some cases, performance is affected by the decrease in operating frequency, while in some cases it is irrelevant. The energy consumption is calculated using the formula of the instantaneous power. Power is proportional to the frequency and voltage squared [6].

$$p = cfv^2 + p_{static} \tag{1}$$

3. Database Selection

We built our database from experimentation. Each experiment is composed of a set of trials, in each trial we analyze the impacts of application of some algorithm. Each algorithm do some combination in processor frequency scaling trying to minimize energy consumption, get better performance or both. We repeated each experiment 30 times and calculated statistics variables like mean and standard deviation of samples.

Table 1. Experiment Trials

	GT-9305T	GT-I9505
OnDemand	S3-OD	S4-OD
Performance	S3-PE	S4-PE
PowerSave	S3-PS	S4-PS
UserSpace	S3-US	S4-US
PegasusQ	S3-PQ	–
Interactive	–	S4-IT
700MHz Fixed	S3-700	–
702MHz Fixed	–	S4-702

In the smartphone we have a lot of dynamic variables inside the operational systems like: GPS, Data Network (WiFi or Cellular network), Calls, Applications updates, Events and the Process from the operational system itself. All the components that can be disabled will be, all the others that is impossible to disable, we will try to minimize the impact in the output variable and control that impact. Experiments will run in smartphones: Samsung Galaxy S3 (GT-9305T) and Samsung Galaxy S4 (GT-I9505). The experiments combinations are in Table 1.

4. The Research Method

Research Question 1: Who processor frequency scaling algorithm consume less energy?

4.1 The Research Design

From the 22 versions of the Android operating system, we choose two of the latest versions (*Jelly Bean and Kitkat*) to do the experiments. This versions cover about 79% from the total of users that use Android smartphones [14] [4].

Response Variable: Energy consumption.

State Variable: Processor frequency scaling algorithm.

Control Object: The same test are used but with default configuration, without applying the technique. Default Governors Android are OnDemand, PowerSave, Performance, UserSpace, PegasusQ for Samsung Galaxy S3 only and Interactive for Samsung Galaxy S4 only.

To make assumptions about the experimental results is necessary to know about the samples data distribution. Firstly, we will plot the points in a graph to explore visually the distribution. The frequency distribution (histogram) is one of the most used evidences to normality, but for a small quantity of samples, histogram can not be a good choice[1], in addition to it, we will use the QQ-Plot graphs and Box Plots to test normality. With this tools is possible to have indicators if the analysed distribution is close to normal distribution.

After the analysis using QQ-Plot graphics, we will try to explain that our distribution can be approximated to a normal using some statistical tests. Shapiro-Wilk[12], Kolmogorov-Smirnov (K-S) test[12] and Lilliefors corrected K-S test, among others, [12][10] can be used to demonstrate the adherence to a normal distribution. The K-S test is an empirical distribution function (EDF) in which the theoretical cumulative distribution function of the test distribution is contrasted with the EDF of the data [12]. A limitation of the K-S test is its high sensitivity to extreme values; the Lilliefors correction renders this test less conservative [10]. It has been reported that the K-S test has low power and it should not be seriously considered for testing normality [9]. In our analyses, we will use Lilliefors test, Shapiro-Wilk test and Anderson-Darling test.

Confirmed that our data are approximately symmetrical, we will apply the paired t student test, otherwise, we will use non-parametric paired Wilcoxon-Mann-Whitney test. We choose to use paired tests because our collected data for all algorithms tested was done in same conditions.

We will combine analysis using a worst, medium and best case of energy consumption with no algorithm to do processor frequency scaling, it means, we will compare the frequency fixed that had the worst energy consumption with all frequency scaling algorithms and using the frequency that are approximately the mean energy consumption and the best case where we have the frequency that have the minimal energy consumption.

Using the tests exposed before, we can answer our research question. We can indicate who algorithm is the best to minimize energy consumption in the analyzed context with a minimum confidence interval.

5. Exploratory Analysis

This section presents...........

5.1 Descriptive Statistics

- media
  - mediana
  - boxplots / qqplots / histogramas

5.2 Adherence Tests

- tabela com os testes realizados. KS, Lilliefors, Anderson Darl, Shapiro, etc.

6. Results Analysis

6.1 Experiment One. YouTube Video Stream with Processor Frequencies

This experiment is characterized by execution of a video stream on YouTube application during 120 seconds, video was selected previously in a rank of the most watched videos at YouTube and the video URL link inserted on proposed environment Web interface.

6.2 Experiment Two. YouTube Video Stream with Governors

In this experiment, the energy consumption collected using standard Android governors (OnDemand, Performance and PowerSave) and PegasusQ for Samsung Galaxy S3 and Interactive for Galaxy S4, are compared with the fixed frequency that had lower energy consumption in the previous experiment.

The chart on Figure 1 contains the energy values for the smartphone Samsung Galaxy S3. The vertical axis indicates the power consumption while the horizontal are governors (frequency switching defined by algorithm) or fixed frequency (no frequency switching). It is noticeable that the lowest energy consumption belongs to the fixed frequency of 700MHz (700MHz-4C in graphic) with four active cores. The governors PegasusQ and Performance present the energy consumption closer to the minimum and nearly equal values with each other. This approximation is given by the constant need for processing in video applications, while the OnDemand got a higher consumption. This can be justified by the high rate of switching frequencies that occur in the device with the operating governor.

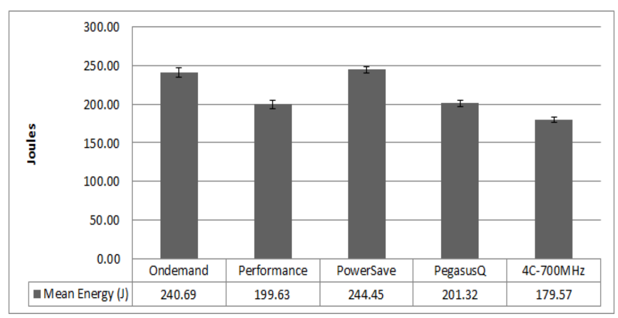

Figure 1. Energy Consumption of Samsung Galaxy S3 with different Governors.

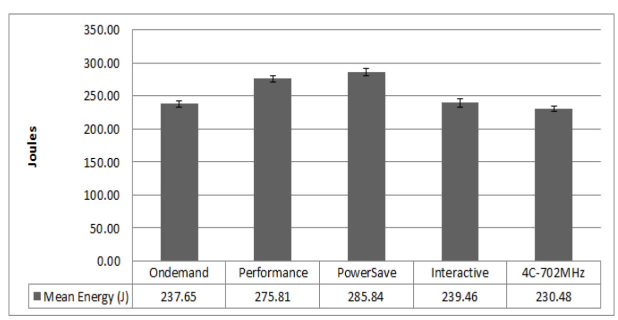

Figure 2. Energy Consumption of Samsung Galaxy S4 with different Governors.

In the graph of Figure 2 are the values of energy collected from video stream experiment with the main governors of Samsung Galaxy S4. The lower energy consumption belongs to a fixed frequency of 702MHz, OnDemand and Interactive with very closer values and Performance with the higher energy consumption.

Do a statistical analysis is more recommended than just compare mean values directly. We will test if data distribution can be approximated to a normal, after that, apply the correct test to compare two samples, student t test for distributions that can be approximated to a normal or Wilcoxon-Mann-Whitney for unknown data distribution.

The energy results from the other governors came out as expected, the Interactive which is the standard in Samsung Galaxy S4 took second lower consumption while the OnDemand was third. The PowerSave showed the largest energy consumption and Performance is the second most energy hungry. PowerSave should be the governor that use less energy, but while it reduce the processor frequency operation, it is necessary to spend more time using this lower frequency, it affect directly the energy consumption.

To analyze data distribution, we will analyze graphically the data. From the QQ-plot graphs from Figure 3 for Samsung Galaxy S3 and for Samsung Galaxy S4 in Figure 4, we can see some normality, but not for all samples, for example, in Figure 3 we can assume that, for Galaxy S3 using OnDemand governors (S3-OnDemand) and using PowerSave governor (S3-PowerSave), this distribution follow a normal distribution, but not for the others governors (S3-Performance and S3-PegasusQ).

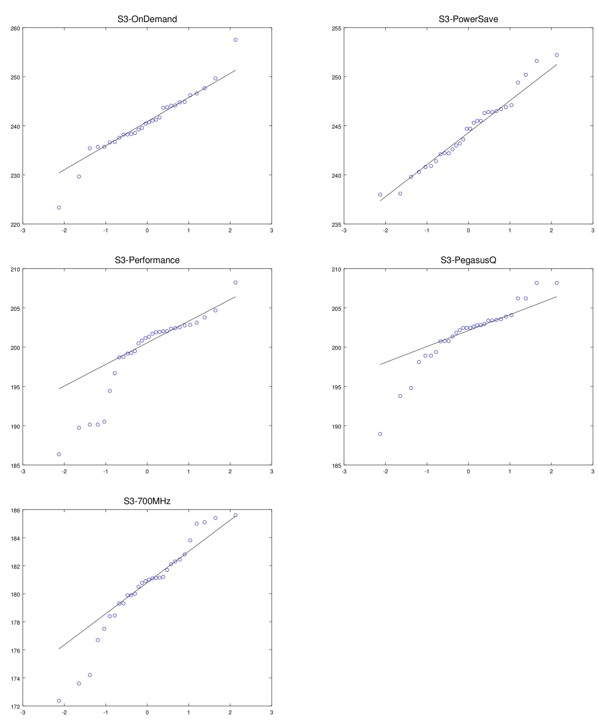

Figure 3. QQPlots Graphics for S3 governors and 700MHz frequency.

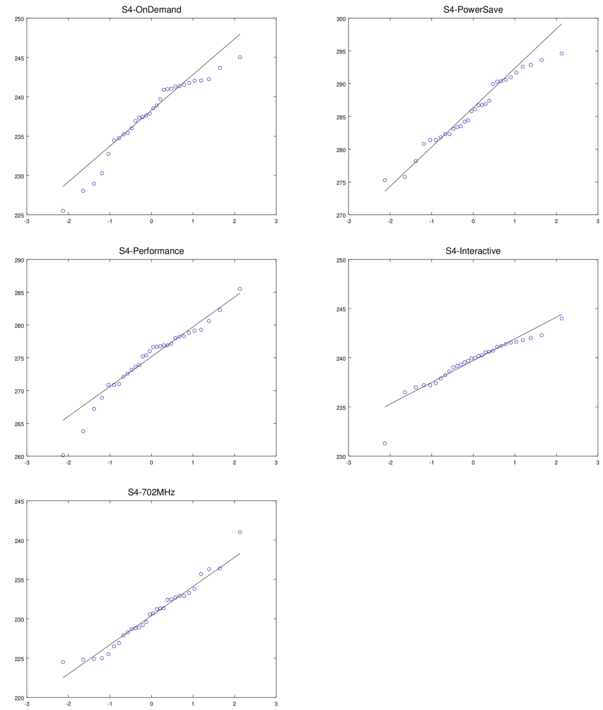

Figure 4. QQPlots Graphics for S4 governors and 702MHz frequency.

In Figure 4 we can assume that when using Interactive (S4-Interactive), Performance (S4-Performance) and PowerSave (S4-PowerSave) can be approximated to a normal distribution as well as using 702MHz fixed frequency (S4-702MHz). The others plots show many outlier points that are distant from normal line but can follow a normal distribution too.

To assume or not our thoughts that who follow a normal distribution, Lilliefors test was applied. The Table **??** shows the results. Only the governors Performance and PegasusQ in Samsung Galaxy S3 do not follow a normal distribution, the others distributions can be approximated to a normal.

It can be seen in the experiments that the optimal frequency to provide energy efficiency to the video stream functionality is equal or closer to 700MHz with the maximum number of active cores, the results are valid for both, Samsung Galaxy S3 and Samsung Galaxy S4. Probably this characteristic can be generalized to other smartphones, but it will require more tests on target devices to prove the hypothesis. The fixed frequency of 702MHz with four active cores (4C-702MHz in graphic) had the lowest energy consumption compared to other governors.

To prove our hypotheses with a minimal confidence interval, student t test were applied to distributions that follow a normal and Wilcoxon to others, results are in Table 4. The table shows the applied test (t student or Wilcoxon) between Samsung Galaxy S3 using 700MHz fixed frequency (S3-700MHz) and the standard governors and Samsung Galaxy

Table 2. Normality Tests for Samsung Galaxy S3

	Lilliefors		Shapiro Wilk		Anderson Darling	
Algorithm	P-Value	∼ Normal	P-Value	∼ Normal	P-Value	∼ Normal
OnDemand	0.21	X	0.2109	X	0.19	X
Performance	0.000768		0.0009073		3.791e-05	
PowerSave	0.6164	X	0.6321	X	0.658	X
PegasusQ	0.01786		0.0008653		0.0002398	
700MHz	0.195	X	0.09014	X	0.09208	X
1100MHz	0.2782	X	0.1943	X	0.1491	X
1400MHz	0.2689	X	0.02742		0.05013	X

Table 3. Normality Tests for Samsung Galaxy S4

	Lilliefors		Shapiro Wilk		Anderson Darling	
Algorithm	P-Value	∼ Normal	P-Value	∼ Normal	P-Value	∼ Normal
OnDemand	0.08668	X	0.0426		0.04047	
Performance	0.2072	X	0.08294	X	0.05182	X
PowerSave	0.3423	X	0.4645	X	0.5096	X
Interactive	0.5022	X	0.01585		0.1072	X
702MHz	0.9548	X	0.3691	X	0.6601	X
1134MHz	0.01896		0.003784		0.001845	
1890MHz	0.652	X	0.1346	X	0.4493	X

S4 fixed frequency of 702MHz compared to his standard governors. The plus signal ('+') implies that the mean energy of fixed frequency can be compared statistically with the others governors energy mean. Distributions that do not follow a normal were applied Wilcoxon test, only for Samsung Galaxy S3 using Performance and PegasusQ.

Can be seen that in our experiments, always a fixed frequency near 700MHz is more energy efficient than standard governors in Android. It was proved with a prediction error of 0.05.

Table 4. Equality Means

	P-Value	Student's t	Wilcoxon
S3-700MHz			
OnDemand	7.75556e-048	+	
Performance	1.7344e-006		+
PowerSave	2.65492e-058	+	
PegasusQ	1.7344e-006		+
S4-702MHz			
OnDemand	4.70731e-008	+	
Performance	5.19865e-042	+	
PowerSave	1.45408e-047	+	
Interactive	2.00429e-015	+	

The technique that not have switching between the frequencies (keep fixed frequency) allowed a 25.39% energy savings for the Samsung Galaxy S3 compared to OnDemand governor and 10.80% compared to PegasusQ governor. While in Samsung Galaxy S4, the energy save was up to 16.43% compared to Performance governor and 3.75% compared to Interactive governor. All the energy savings percentage are in Table 5.

Table 5. Energy Savings

	Energy Saving %
S3-700MHz	
OnDemand	25.39%
Performance	10.04%
PowerSave	26.54%
PegasusQ	10.80%
S4-702MHz	
OnDemand	3.01%
Performance	16.43%
PowerSave	19.36%
Interactive	3.75%

6.3 Research Impacts

In academic studies, new algorithms to do frequency scheduling have been applied using different areas like: Machine Learning and Neural Networks, Linear and Logistic Regression, among others. To assume that a new proposed algorithm technique is better than another is necessary do experimentation, our proposal is to offer an integrated approach to measure and analyses energy consumption in mobile devices. This approval is one step to export this algorithm to industry and incorporate in new smartphones, doing them more energy efficient.

7. Concluding Remarks and Future Works

Given the need for an environment which would help to soften the problem of fragmentation and permit energy analysis, the proposed study offered a combination of hardware and software which assists energy characterization tests on multiple devices through a Web interface.

To test the proposed environment, three macro experiments covering playing video streams, processor load, frequencies scaling and energy governors analysis were done. The first and second experiment showed that processor frequencies close to 700MHz for playing videos on stream are energy efficient. This information can permit energy savings up to 26% compared with the Android governors standards in the smartphones tested. In processor load experiment, it was possible to indicate the use of the maximum number of cores in the maximum frequency when there is CPU-bound processes, this information can provide energy savings in the execution of processes that make heavy use of processing.

For future work, we need to analyze more scenarios to identify the optimal frequency on each one. This knowledge will be applied in the development of a new governor for the Android operating system.

Another line of evolution is the availability of the environment as a service through Web interface and availability of source code for academic community. Researchers will be able to characterize the energy consumption of their techniques on real devices.

References

[1] F. A. Discovering statistics using spss. *3 ed. London: SAGE publications Ltd*, page 822, 2009.

[2] F. Q. B. S. A. L. O. C. J. A. L. M. S. P. H. R. C. R. C. G. M. A. G. Silva-Filho, P. T. L. Bezerra. Energy-aware technology-based dvfs mechanism for the android operating system. *In Proceedings of Brazilian Symposium on Computing System Engineering (SBESC)*, 2012.

[3] E. K. Atkinson. An introduction to numerical analysis (2nd ed.). *New York: John Wiley and Sons, ISBN 978-0-471-50023-0*, 1989.

[4] A. Central. Lollipop closes in on 10 percent of active android installs, kitkat and jelly bean each fall under 40 percent. *Available in: http://www.androidcentral.com/lollipop-closes-10-percent-active-android-installs-kitkat-falls-under-40-percent. September - 21 - 2015*, 2015.

[5] A. Developer. Android, the world's most popular mobile platform. *Available in: http://developer.android.com/about/index.html. Access in March - 02 - 2015*, 2015.

[6] M. Domeika. Software development for embedded multi-core systems: A practical guide using embedded intel architecture. *Newnes, Newton, MA,*, 2008.

[7] T. M. R. C. L. J. Y. Z. Q. L. F. Xu, Y. Liu. Optimizing background email sync on smartphones. *In Proceeding of the 11th International Conference on Mobile Systems, Applications and Services, MobiSys 13, pages 5568, New York, NY, USA, ACM*, 2013.

[8] G. S. W. K. J. W. G. P. Perrucci, F. H. P. Fitzek. On the impact of 2g and 3g networks usage for mobile phones' battery life. *in European Wireless Conference*, 2009.

[9] T. HJ. Testing for normality. *New York: Marcel Dekker*, 2002.

[10] P. J and B. B. Medical statistics: A guide to data analysis and critical appraisal. *Blackwell Publishing*, 2005.

[11] Z. M. S. S. O. S. J. Huang, F. Qian. Screen-off traffic characterization and optimization in 3g/4g networks. *In Internet Measurement Conference (IMC'12)*, 2012.

[12] E. A. Oztuna D and T. E. Investigation of four different normality tests in terms of type 1 error rate and power under different distributions. *Turkish Journal of Medical Sciences*, pages 171–176, 2006.

[13] A. S. R. Palit, K. Naik. Impact of packet aggregation on energy consumption in smartphones. *In Proceedings of the 7th International Wireless Communications and Mobile Computing Conference pp. 589-594, Istanbul, Turkey*, 2012.

[14] O. Signal. Android fragmentation visualized july 2013. *Available at: ¡http://opensignal.com/reports/fragmentation-2013/¿, accessed in June, 22*, 2015.

[15] Statista. The statistics portal. *Available in: http://www.statista.com/statistics/278305/daily-activations-of-Android-devices. Access in November - 24 - 2014*, 2014.

[16] A. S. V. Pallipadi. The ondemand governor: past, present and future. *In Proceedings of Linux Symposium, vol. 2, pp. 223-238*, 2006.