What is Eating Up Battery Life On My SmartPhone: A Case Study

Grace Metri[†*], Abhishek Agrawal[*], Ramesh Peri[*], and Weisong Shi[†]
[†] Department of Computer Science, Wayne State University, Detroit, Michigan
[*] Software Services Group, Intel Corporation, Santa Clara, California
gmetri@wayne.edu, {abhishek.r.agrawal, ramesh.v.peri}@intel.com, weisong@wayne.edu

Abstract— **Smartphones emerged as the new necessary gadget to many. A smartphone can combine some or all functionalities of several other devices such as a personal computer, phone, personal game console, music player, radio, and/or GPS. Unlike most of the above listed technologies which are switched ON on a need-to basis, a smartphone is always ON.**
Since a smartphone can run background tasks even during idle mode and since it is limited by its battery life, it becomes necessary to understand what really happens in the background and how it affects the battery life and consequently how to improve it. To this end, we analyzed two smartphone platforms specifically looking at how the energy consumption varies depending on the background applications and network connection type. For instance, we show that you can increase the energy efficiency of an iPhone by up to 59% when streaming music on Wi-Fi as opposed to 3G. Also, when the phone is using 3G, we show that network applications running in the background can reduce the energy efficiency of an iPhone by up to 72% when compared to real idle state. Our observation sheds light on what is eating up the battery life of a smartphone and led us to provide optimization techniques to increase the battery life.

Keywords-component; Energy Efficiency, Wireless Communication and Smartphones.

I. INTRODUCTION

Smartphones, owned by over 45.5 million people in the United States, are the fastest growing segment of mobile devices [1]. It is forecasted that by 2015, smartphone users will increase worldwide to over 1.5 billion and smartphone sales volume will reach 448.8 million while Notebook PCs (Microsoft and Mac) will reach 260 million [2]. Smartphones' increasing popularity stems from their capability to run numerous types of applications ranging from simple ones such as playing music to very sophisticated ones such as group gaming. Despite the faster CPUs and networks, and larger memory, still the utility of these phones are limited by their battery life. As a result, energy efficiency of smartphones is a forefront area of study in the mobile field.

In this paper, we refer to smartphone's idle time to the case where the screen is off while there might be applications running in the background which are not necessarily used by a smartphone user. This idle case is common to many users who are warned in some cases by the smartphone providers about the impact of idle applications. However, there is a lack of understanding of the exact impact and how to reduce it in the event of needing those applications in the background. There are several works studying battery life during runtime mode but little has been paid to idle time. Energy consumed is a function of the average power consumed over time multiplied by time. Therefore, since the smartphone remain for extended periods in this mode such a study becomes necessary to increase the battery life of smartphones without necessarily limiting their key feature of multitasking.

To this extend, this paper makes the following contributions:

- We provide two detailed case studies of two popular smartphone platforms: an iPhone and an Android where we observed the impact of background applications and network connection type on the CPU utilization and energy consumption of the devices.
- The focus of current research literature on power profiling of smartphones is focused on Android due to its open-source nature while leaving out its number one competitor, the iPhone. Our detailed iPhone study reduces the current research gap.
- We provide possible optimization techniques to increase the energy efficiency of smartphones despite the presence of background activities.
- We show that even though some concepts are widely known to increase energy efficiency of smartphones such as avoiding polling functions and replacing them with event driven functions, in addition to coalescent network activities especially for not-actively used systems, still the two most popular platforms have not adopted all these techniques. Therefore, there is still potential of improvement.
- We aim to increase the awareness of what is eating up the smartphone's battery life for the users and help them reduce it using the information presented in this paper.

The remaining paper is organized as follows. We present related work in Section 2 followed by categorizing

978-1-4673-5328-1/12/$31.00©2012 IEEE

the smartphone usage models in Section 3. Section 4 illustrates two case studies of popular smartphone platforms: an iPhone and an Android. Section 5 highlights optimization techniques. Then, we conclude in Section 6.

II. Related Work

There are several work focused on different aspects of energy efficiency of smartphones:

- **Wi-Fi sensing and tethering:** Kim et al. [3] introduced WiFisense, a Wi-Fi sensing system which maximizes the use of Wi-Fi access points while improving the energy efficiency through adaptive scan-triggering time intervals. Another Wi-Fi related research is Wi-Fi tethering which refers to the use of the Wi-Fi interface of a smart phone as a mean to share its own Internet connection with other clients such as tablets, smartphones, or computers. DozyAP [4] is a system designed to put the Wi-Fi interface of a smartphone, which is acting as a mobile software access point, into timed and client-approved sleep mode in order to increase its energy efficiency. Similarly to this work, we studied the potential of reducing energy consumption when using Wi-Fi during idle mode as opposed to their research which focus on Wi-Fi during active mode.
- **Profiling tools:** There are several tools available such as [5-6] designed to profile smartphone's application performance, energy efficiency, and/or network impact in an effort to give insights to developers to improve the resource utilization of their applications. Zhang et al. [14-15] introduce PowerTutor which is a power monitoring application for Android users in order to monitor the behavior of applications on their devices. Our work focuses on using current profiling technology to understand the platform behavior.
- **Radio resource allocation for 3G networks:** Qian et al. [7] characterizes the impact of operational state machine settings of 3G networks and provides insights on the present inefficiencies which are due to the interplay between inefficiencies which are due to the interplay between the device's applications and the state machine behavior. Then, they propose an optimal state machine setting which can, for instance, reduce the energy of streaming YouTube videos by up to 80%.

III. Smartphones Usage Models

A smartphone full potential can only be achieved by its ability to connect to the Internet. The common connection models are through 3G cellular data networks and Wi-Fi, in addition to the recent penetration of 4G LTE link which is currently not supported in all areas. The smartphone usability is diverse and highly dependent on user's demographics. Recent study by [8-9] revealed that the number of applications used varies from 10 to 90 per user and the number of interactions per day varies from 10 to 200.

Smartphones usage models are broadly characterized as follows:

- **Streaming media:** This category of applications provides a mean to watch/upload/download videos or music such as YouTube, Netflix, and Pandora.
- **Social computing:** It refers to applications whose primary role is social interaction such as blogging, social networking, instant messaging, and e-mailing.
- **Informational:** It refers to applications used to retrieve information such as news feeds or search engines.
- **Utility computing**: It refers to applications used to perform a computational task or provide a service such as calculators, calendars, or reminders.
- **Gaming:** It can either refer to local games on the devices or network games where multiple players can play together over the network.

IV. Case Studies

This section illustrates two case studies using two different popular platforms: an iPhone and an Android. For each platform, we considered different background scenarios and evaluated their impact on the devices.

A. *Case 1: The iPhone*

- **Experimental Setup and Methodology:** During our experiments, we used an iPhone 4S running iOS 5.1.1. We collected our data using Apple's Instrument which allows dynamic profiling of different performance metrics of the iPhone [10]. We collected the data at one second intervals. We performed Test 1 through Test 4 as listed in table 1. Depending on the test type, we started the corresponding applications and let it run in the background. Then, we started the profiler on the phone. Next, we turned the display off. Upon the completion of the test period, we stopped profiling. Lastly, we connected the phone to a computer via USB, retrieved the data, and analyzed the results.

TABLE 1: TYPES OF TESTS

Test Name	Test Description	Purpose	Applications
Test 1	Mix of application with different network utilization	Impact of constantly streaming data in the background.	Pandora Skype LinkedIn Dropbox Facebook
Test 2	Mix of application with different network utilization	Impact of intervallic network connected applications	Skype Facebook LinkedIn Dropbox
Test 3	Mix of utility and non-network applications	Comparing non-network applications to network applications	Calculator Puzzle Game Roller Coaster Game Units Converter
Test 4	No application	True idle comparison	None
Test 5	Only monitoring Applications	Non-network application for Android	Network Log SystemPanel Pro Battery Monitor Widget Pro

- **Experimental Results:** We focused in this paper on the sleep/wake, energy usage, CPU activity, and network activity profilers of the *Energy* Diagnostics Instrument.
  - ***Sleep/Wake:*** Sleep/wake status instrument is part of the Energy Diagnostics template. It records the devices sleep and wake modes. The iPhone has four major modes: *Sleep*, *Attempting to Sleep, Running,* and *Waking* ordered from the least power consuming state to the most consuming one. Figure 1 and figure 2 represent the results of the sleep/wake modes of test 1 through 4 using Wi-Fi and 3G respectively. State 0, state 1, and state 2 on the y-axis represent *Sleep*, *Attempting to Sleep*, and *Running* states respectively. (Waking requires a negligible amount of time. Therefore, it is omitted from these graphs)

  - ***Energy Usage:*** The Energy usage instrument does not provide an exact usage value of power; however, it has a scale from zero to twenty ranging from the most efficient to the least. Figure 3 and figure 4 represent energy usage of the iPhone using Wi-Fi and 3G respectively.

  - ***CPU Activity:*** Using the CPU Activity instrument, we were able to collect the total percentage of activity over time in addition to each application's activity status change such as running or suspended. Figure 5 and 6 represent the percentage of CPU activity using Wi-Fi and 3G respectively.

  - ***Network Activity:*** Using the network activity instrument, we collected the Wi-Fi bytes in and out, in addition to the cell bytes in and out. Figure 7, 8, 9, and 10 represent network bytes in and out using Wi-Fi, using Wi-Fi excluding test 1 (for graph clarity purpose), using 3G, and using 3G excluding test 1 (for graph clarity purpose) respectively

Table 2 summarizes the average CPU activities and the average energy usage for all tests during both Wi-Fi and 3G.

Results of Test 1:

During Test 1, we noticed the following:

- In the case of Wi-Fi and 3G, the device remained in running state throughout the test which is due to the continuous streaming of music.
- The energy usage alternated between lower usages to high usages. By comparing the energy usage to the network activities, we noticed that the energy spikes are consistent with the fetching of new network activities.
- When comparing the network activities of Wi-Fi and 3G, it is evident that the quantity of packets are fewer but larger in size in the case of Wi-Fi when compared to 3G. This explains why Wi-Fi has lower number of energy spikes when compared to 3G. On average, we can save 59% more energy when using Wi-Fi as opposed to 3G.

Results of Test 2:

During test 2, we noticed the following:

- Similar observation as Test 1 regarding the alignment of energy usage to the network activity.
- Unlike Test 1, the device alternated between sleep and running states. Also, in the case of 3G, it remained for longer periods attempting to sleep.
- Percentage of CPU usage is lower by 44.86% when comparing Wi-Fi to 3G.
- The average energy usage is lower by 64% when comparing the Wi-Fi to 3G.

Results of Test 3 and Test 4:

During Test 3 and Test 4, we noticed the following:

- Test 3 and Test 4 have similar results where the device remained in sleep mode for long periods of time. We noticed that the interval of remaining asleep increased over time. We concluded that keeping idle non-network applications does not have a noteworthy effect on the energy efficiency of the device.
- We also observed a counterintuitive fact when we compared Test 3 and Test 4 while Wi-Fi was enabled. Test 4 (which lacks any background applications) had higher CPU activities and higher average energy consumption than Test 3 (which has utility applications running in the background). For further investigation, we connected the USB cable between the iPhone and the computer then ran both tests scenarios while collecting the CPU activities via the *Activity Monitor Instrument*. We noticed the presence of "Backup" function calls which synchronizes the phone with Apple's iCloud when no background applications are present and the phone is in Wi-Fi mode.

  Apple's iOs is configured to perform automatic backup when the iPhone is connected to power, in Wi-Fi mode, and is not running any applications. Our original test results of Test 4 can be explained by the following: since the phone was connected to Wi-Fi and it was in true idle state, the iOS kept performing poll system calls to check if power was connected in order to perform backup. As a result, CPU activities and energy consumption were higher in Test 4 when compared to Test 3.

TABLE 2: AVERAGE CPU ACTIVITY AND ENERGY USAGE

	Test 1	Test 2	Test 3	Test 4
Average CPU Activity – Wi-Fi	8.12	1.02	0.38	0.68
Average CPU Activity – 3G	8.86	1.85	0.42	0.35
Average Energy Usage – Wi-Fi	4.02	1.12	0.36	0.47
Average Energy Usage – 3G	9.85	3.12	0.87	0.86

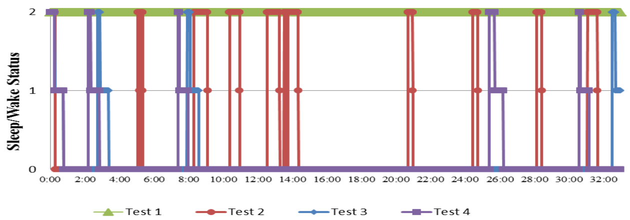

Fig. 1: Sleep/Wake status of iPhone using Wi-Fi

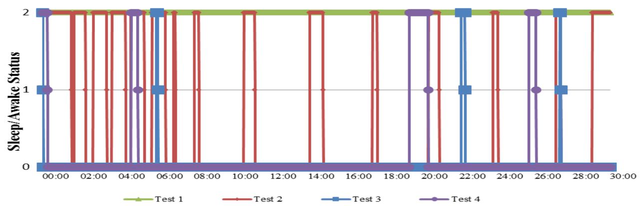

Fig. 2: Sleep/Wake status of iPhone using 3G

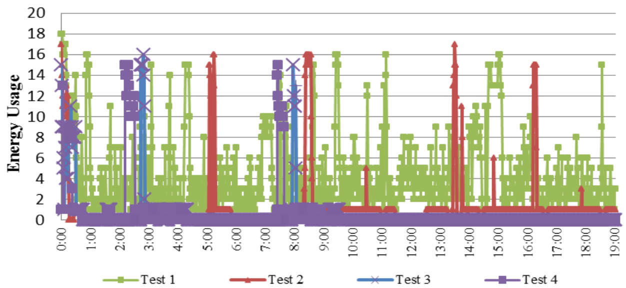

Fig. 3: Energy usage of iPhone using Wi-Fi

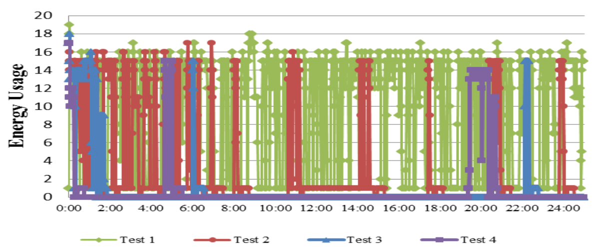

Fig. 4: Energy usage of iPhone using 3G

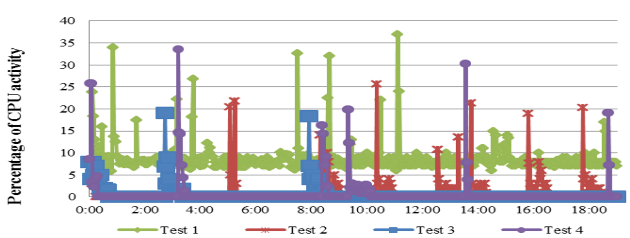

Fig. 5: Percentage of CPU activity of iPhone using Wi-Fi

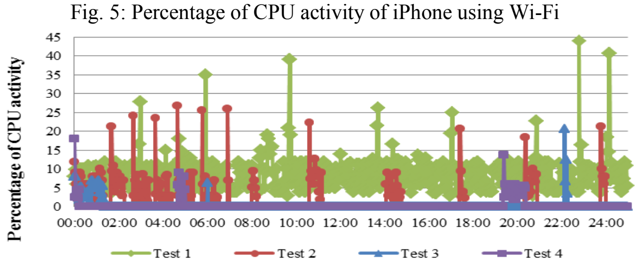

Fig. 6: Percentage of CPU activity of iPhone using 3G

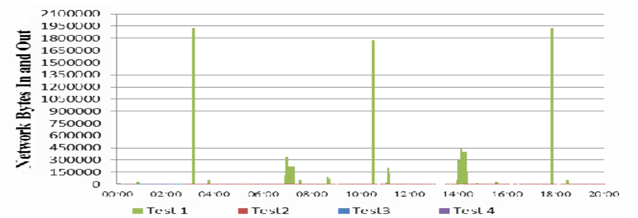

Fig. 7: Network Bytes In and Out of iPhone using Wi-Fi

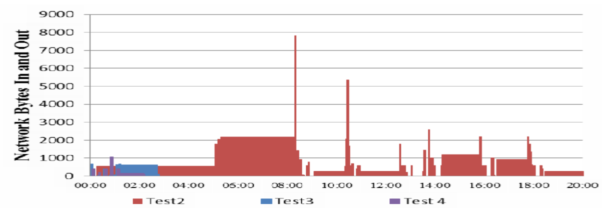

Fig. 8: Network Bytes In and Out of iPhone using Wi-Fi excluding Test 1

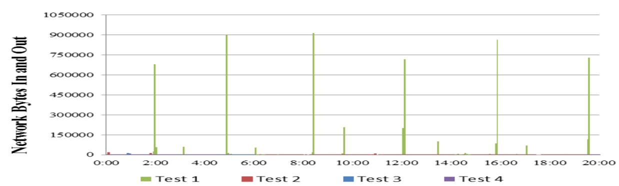

Fig. 9: Network Bytes In and Out of iPhone using 3G

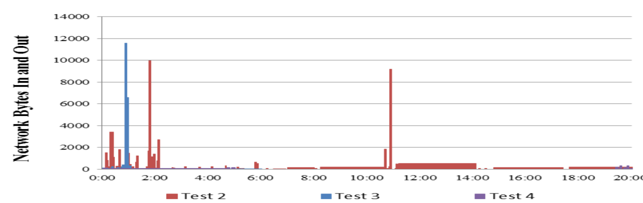

Fig. 10: Network Bytes in and Out of iPhone using 3G excluding Test 1

B. Case 2: The Android

- **Experimental Setup and Methodology:** During our experiments, we used Samsung S3, model number GT-I9300, a rooted Android version 4.0.4, kernel version 3.0.15. We used three different applications to collect our data. The first application is *Network Log* which collects real-time network activity of each application including detailed number of bytes sent by applications and the appropriate timestamp [11]. The second application is *Battery Monitor Widget Pro* which records the utilization in mA, the voltage of the battery in mA, and the battery's temperature [12]. The third application is *SystemPanel App/Task Manager Pro* which records the system usage such as CPU usage for each application and overall CPU usage in addition to other information that were not used in this paper [13]. We performed test 1, 2, and 5 as described in table 1.

- **Experimental Results:** Using the Network Log, we were able to collect the network activity of the device. Figure 11, 12 and 13 represent the sum of network bytes in and out of each individual application running on Android while using Wi-Fi for test 1, 2, and 5 respectively. Similarly, figure 14, 15, and 16 represent network bytes but while using 3G for test 1, 2, and 5 respectively.

Network Usage:

It is evident again that Pandora in test 1 dominated the network usage. There were more network bytes sent/received periodically during 3G when compared to Wi-Fi. Moreover, Skype and Viber periodically send/received packets. NOTE: during test 5, Viber was manually terminated from the task manager and it was not supposed to run in the background; however, despite its termination, it remained active. Only the un-install can prevent its activity.

During our experiments, we uninstalled Viber from the iPhone and reinstalled it. Upon restart, we received the following message as shown in the figure to the left that Viber "will not drain your battery". However, based on our experiments on the Android, we noticed that even when the application is not running in the background, it still periodically sends and receives packets over the network which can change the radio state from sleeping to running and thus utilize the battery.

CPU Utilization:

Figure 17 and 18 represent the percentage of CPU utilization which we collected from *SystemPanel App.* The results are similar to the results of the tests on iPhone where the CPU utilization when comparing Wi-Fi's values to the 3G's.

Energy Usage:

Lastly, we collected the battery consumption from the Battery Monitor Widget. We noticed that on average there is a 9 to 14 percent energy savings when comparing the energy consumption of all the Wi-Fi tests to the 3G tests. There is also 39 to 47% energy savings when comparing the energy consumption between test 5 and test 1.

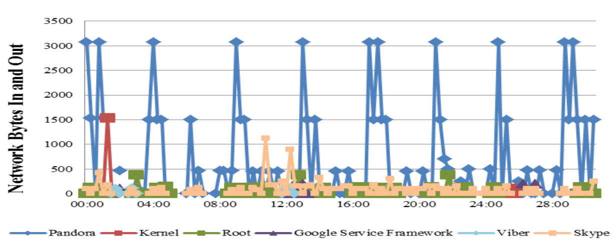

Fig. 11: Network bytes in and out of Android using Wi-Fi during test 1

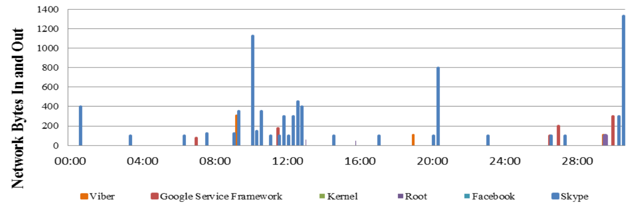

Fig. 12: Network bytes in and out of Android using Wi-Fi during test 2

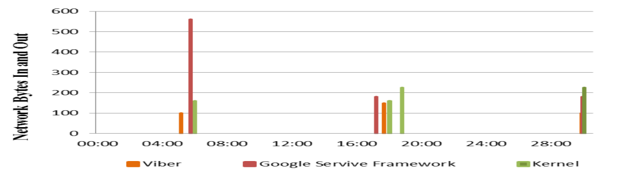

Fig. 13: Network bytes in and out of Android using Wi-Fi during test 5

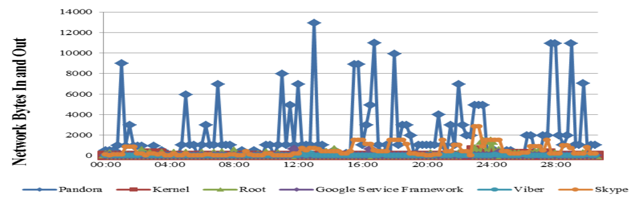

Fig. 14: Network bytes in and out of Android using 3G during test 1

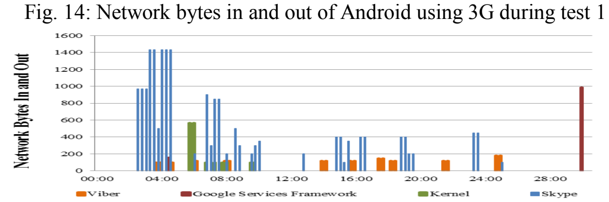

Fig. 15: Network bytes in and out of Android using 3G during test 2

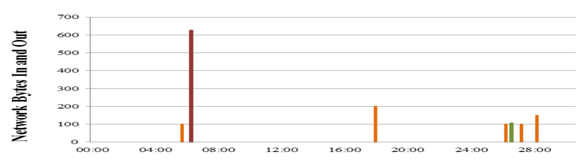

Fig. 16 Network bytes in and out of Android using 3G during test 5

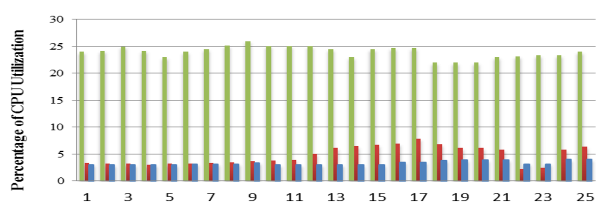

Fig. 17: Percentage of CPU usage of Android using Wi-Fi

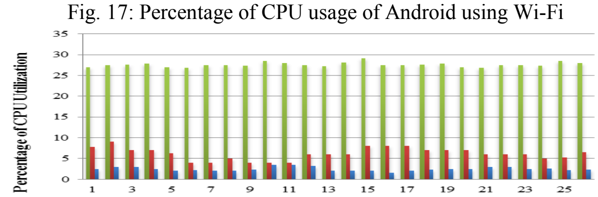

Fig. 18: Percentage of CPU usage of Android using 3G

V. Optimization Techniques

Based on our observation, we derived a list of optimization techniques in order to increase the overall energy efficiency:

1. **Coalesce of network activities:** Every time there is a new network connection, the radio transitions to a full power state and stay in that power state after the transmission is complete [5]. Therefore, it is important to group the network activities as close together as possible, even if they remain in consecutive order, in order to attain longer inactive periods. In particular, we suggest that the network activities performed by the Kernel and Google Service Framework can be coalesce together because these are system level activities and should be timed by the OS.

2. **Improve the policy of scheduled backup:** Backup is a necessary feature given the importance of the information stored on a smartphone. It is a good strategy to perform automatic backups when the following three conditions are met: the phone is in true idle and the phone is connected to Wi-Fi network and connected to an external power source. However, unlike the current implementation where the current iOS keeps polling to check if external power source is connected when the phone is connected to Wi-Fi and is in true idle, automatic backup function should be event driven. In other words, if the phone is connected to external power, then the function checks if the other conditions are present to perform automatic backup.

3. **Keep the NIC and radio in low power states:** Performance during idle state does not require the same performance requirements as the performance when a user is actively utilizing the smartphone. As a result, the exploration of reducing the power states of the NIC and radio during network activity if the I/O is off can further increase the efficiency of a smartphone. Keeping the NIC and radio in low power states are not new concepts. However, the current focus is reduction during active mode but exploring new potentials for idle mode is necessary because the requirements during the latter are different from the requirements during active mode.

4. **Informed freedom:** There are settings in smartphones such as the Android to limit the number of background applications running. However, based on our experiments, not all applications are created equal. For instance, utility applications do not reduce the efficiency of a smartphone. Therefore, instead of putting a cap on the number of applications running by having them automatically forced to end, there should be awareness of which applications and network types can reduce the efficiency of the smartphone and which ones do not. Thus, the user can act accordingly.

VI. Conclusion

This paper evaluates the impact of background application on the battery life of a smartphone. We showed on two different platforms, the iPhone and Android platforms, that using Wi-Fi as opposed to 3G will decrease the consumption of energy of the smartphone and thus make it more energy efficient. We also showed how the network activities (packet size and interval between packets sent/received) directly affect the energy consumption and ultimately battery life. Finally, we aim for our findings to be used by smartphone's users in order extend the battery life of their devices and for our recommendations of coalesce of network activities, improving the policy of scheduled backup, and keeping the NIC and radio in low power states be adopted by the platform and/or OS providers.

References

[1] A. Pathak, Y.C. Hu, M. Zhang, P. Bahl, Y. Wang. "Fine-Grained Power Modeling for Smartphones Using System Call Tracing," in *Proceedings of the sixth conference on Computer systems* (EuroSys '11), 2011.

[2] http://www.researchandmarkets.com/research/7a1189/worldwide_smartpho

[3] K.H. Kim, A. W. Min, D. Gupta, P. Mohapatra, J. P. Singh. "Improving Energy Efficiency of Wi-Fi Sensing on Smartphones," In INFOCOM, 2011.

[4] H. Han, Y. Liu, G. Shen, Y. Zhang, Q. Li. "DozyAP: power-efficient Wi-Fi tethering," in *Proceedings of the 10th international conference on Mobile systems, applications, and services* (MobiSys '12), 2012.

[5] A. Pathak, Y. C. Hu, M. Zhang. "Where is the energy spent inside my app? Fine Grained Energy Accounting on Smartphones with Eprof," in *Proceedings of the seventh ACM European conference on Computer systems* (EuroSys '12), 2012.

[6] F. Qian, Z. Wang, A. Gerber, Z.M. Mao, S. Sen, O. Spatscheck. Profiling Resource Usage for Mobile Applications: A Cross-layer Approach. In MobiSys, 2011.

[7] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, D. Estrin. "A First Look at Traffic on Smartphones," in *Proceedings of the tenth ACM SIGCOMM conference on Internet measurement* (IMC '10), 2010.

[8] F. Qian, Z. Wang, A. Gerber, Z. M. Mao, S. Sen, O. Spatscheck. "Characterizing Radio Resource Allocation for 3G Networks," in *Proceedings of the tenth ACM SIGCOMM conference on Internet measurement* (IMC '10), 2010.

[9] H. Falaki, R. Mahajan, S. Kandula, D. Lymberopoulos, R. Govindan, D. Estrin. "Diversity in Smartphone Usage," in *Proceedings of the eighth international conference on Mobile systems, applications, and services*, (MobiSys '10), 2010.

[10] https://developer.apple.com/library/mac/#documentation/developertools/conceptual/InstrumentsUserGuide/Introduction/Introduction.html

[11] http://code.google.com/p/iptableslog/

[12] http://www.3c71.com/android/?q=node/1#main-content-area

[13] http://android.nextapp.com/site/systempanel

[14] Z. Yang, "PowerTutor – A Power Monitor for Android-Based Mobile Platforms" EECS, University of Michigan, retrieved September 2, 2012. Available: http://ziyang.eecs.umich.edu/projects/powertutor/

[15] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. Dick, Z. M. Mao, L. Yang. "Accurate Online Power Estimation and Automatic Battery Behavior Based Power Model Generation for Smartphones," *in Proceedings of the eighth IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis* (CODES/ISSS '10), 2010.