Modelling and Improving the Battery Performance of a Mobile Phone Application: A Methodology

Edip Demirbilek, Jean-Charles Grgoire
Institut National de la Recherche Scientifique
Montral, QC, CANADA H5A 1K6
+1 (514) 929-8318
{edip.demirbilek, gregoire}@emt.inrs.ca

Ahmad Vakili, Lionel Reyero
Summit Tech Inc.
Montral, QC, CANADA H2N 1N2
+1 (514) 466-9560
{vakili, lionel}@summit-tech.ca

***Abstract*—We evaluated the power consumption of an industry-grade IMS/RCS application that periodically updates and retrieves the availability status as well as inbox query task to/from the server over Wi-Fi interface. Our aim was to find out the amount of additional energy consumed for each unique tasks. We modelled the battery usage of always-on features and identified each feature's contribution to the whole consumption. While presenting the work we also paid attention to the process we went through and provided a methodology for similar future work.**

I. Introduction

There has been a reasonably large amount of research on modelling and improving the battery consumption of mobile phones. However, some works have become less relevant due to changes in mobile platforms while some other works are too specific or too complicated to fit in the usual application development process. As a result, improving the battery performance of an application still requires a notable amount of work, investment and hands-on experience.

Our research took an always-on application as a use case in order to model the battery usage of always-on features and identify each feature's contribution to the entire consumption. We evaluated the power consumption of an industry-grade IMS/RCS application [1] that periodically updated and retrieved the availability status as well as inbox query task to/from the server over Wi-Fi interface using Samsung mobile phones. Each time interaction with the server was required, the Wi-Fi interface was activated for transmission. Our aim was to find out the amount of additional energy consumed for each unique task during these Wi-Fi interface utilization and to model the overall battery consumption of the application, which run as an always-on background process on the mobile phone [2]. While presenting the works done in detail we also paid attention to the process we went through and provide a methodology for similar future works.

The process we followed can be summarized as follows: First of all, we stripped down the IMS/RCS application to isolate and model the battery usage of each feature –registration, presence on and IMAP query– and developed several small clients each representing only one of the atomic SIP or IMAP protocol operations.

We determined the amount of energy consumed for each atomic operation employing an external power monitor for accurate detailed measurements. We then modeled the overall battery consumption of the application given specific timer settings since we were able to set the frequency and the count of these atomic operations for each feature.

With these findings we obtained the optimum timer settings for each feature in order to decrease IMS/RCS application contribution to the battery consumption to as low as possible while still providing a good user experience. In addition to that we presented the effect of various timer settings on the battery lifetime in comparison to the mobile phone factory IDLE state.

Later in this paper, we present a more detailed methodology followed to execute each of these steps mentioned above.

II. Related Work

The first group of works we reviewed concentrate on specific use cases of mobile phones. Chen et al. [3] look into the energy consumption of smartphone display applications, while Constandache et al. [4] investigate energy efficient localization and Nurminen [5] investigate how much energy different communication alternatives consume.

Perrucci et al. [6] present the results of power and energy consumption measurements conducted on mobile phones for 2G and 3G networks. Similarly, Havarinen [7] concentrates on 3G network interfaces while analyzing how keep-alive messages influence battery lifetime.

There is also significant research about improving the battery performance at the operating system level. Schulman et al. [8] propose an approach for energy-aware cellular data scheduling. A similar work done by Kononen et al. [2] suggests timer alignment methods for reduction of energy consumption.

Building the test setup and conducting the experiments are another big issue researched by many. Zhang et al. [9] present a tool that uses built-in battery voltage sensors. Likewise Murmuria et al. [10] implement a system that makes use of the per-subsytem time shares reported by the operating system's power management module. There are also hybrid approaches like Bornholt et al. [11] that use both external power monitors as well as on board power sensors for measuring the battery consumption.

Balasubramanian et al. [12] look into the tail energy in GSM, 3G and Wi-Fi and present a protocol that reduces energy consumption considering the tail energy. These authors' detailed discussion of tail energy timeouts have been valuable inputs to our research.

We provide a methodology without deploying any specific script or software that is intended to trace system calls and require sophisticated analysis afterwards. Our approach provides a more practical and doable method for application developers employing only their existing know-how about their system and utilizing only external tools for accurate and dependable results.

III. Tail Energy

Wi-Fi is clearly more energy efficient than 3G and GSM when transferring the data across the network [5] [12]. That difference is determined by the amount of energy drained during the data transfer and during the high-power states after the transfer is completed. The latter factor bears a big role since mobile phones turn on the radio interface for every small amount of periodic network data exchange. In GSM and 3G, the radio interface stays in high-power states for some time even after the network activity has ended. This additional time spent in high-power state is called the tail time and additional energy consumed is called the tail energy. Balasubramanian et al. [12] measured the tail time for GSM as 6 secs and for 3G 12.5 secs.

In Wi-Fi the transition between the power states occurs differently. Typically Wi-Fi enters to the idle state much more rapidly than the 3G due to built-in PSM mechanism after data transfer is completed. 3G and GSM radio circuitry continue powered on if the next utilization is within their tail time while Wi-Fi is able to enter low-power sleep state between data packets for shorter time intervals [5]. Researchers didn't observe a specific duration for the tail state of Wi-Fi but witnessed a monotonic increase in tail energy while testing for different timeout values. It is also worth noting that there is a high maintenance overhead energy for keeping the Wi-Fi interface on [12].

From these results it is obvious that any power consumption model involving periodic network data transfer has to consider the tail energy and set the periodic timers accordingly.

Executing a network data transfer activity while there is another parallel connection will certainly reduce the energy consumption as well. We measured the amount of tail energy for each operation independently and therefore we know in real life scenarios mobile phone battery performance is at least as good as measured results. Grouping these operations with clever timeout settings on the application layer will certainly reduce the overall power consumption and result in longer mobile phone standby duration.

IV. Measurement

A. Power Evaluation Setup

We followed the GSM Association Battery Life Measurement and Current Consumption Technique [13] while building our test setup and conducting the tests.

We performed the energy measurement with the Agilent[1] 6632B [14] high performance DC power supply. We measured the amount of current being drained by the mobile phone with a Matlab script using industry standard SCPI commands via the GPIB interface. Integration to the Matlab environment was done using the simplified VXI Plug&Play drivers. This configuration enabled us to easily and accurately monitor the output current being drained.

We used Summit-Techs next generation IMS/RCS client [1] on smartphones using GSMA Rich Communication Services. We installed the modified clients on Samsung Android installed mobile phones. Figure 1 depicts the hardware and software components and the connections between them as configured during the tests.

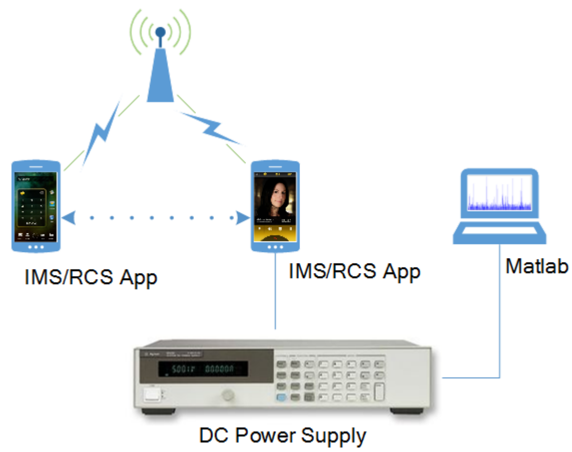

Fig. 1. Power measurement setup connecting DC power supply, mobile phone under test, Wi-Fi access point and the Matlab script measuring the energy consumption over the GPIB interface.

Network data transfer was tested over the Wi-Fi interface. During the tests the mobile phone was connected to a dedicated or shared access points in each case and the results obtained for both cases compared. The phone was stationary during the tests.

The mobile phone battery was replaced with a dummy battery fixture to enable accurate current measurements while powering the mobile phone from the DC Power Supply. That configuration provided the current for any active battery management functions within the mobile phone while enabling to perform measurements from the Matlab environment. We 3D printed a dummy battery having the same dimensions as the original battery. This provided a safer environment and we also found it to be more practical and stable during the tests. Figure 2 shows that 3D printed component.

Fig. 2. 3D printed dummy battery fixture placed in the mobile phone battery slot with the power cable connected to the DC power supply output interface.

[1]As of 1 November 2014 Agilent's Electronic Measurement business is now Keysight Technologies.

B. Measurement Methodology

The GSM Association Battery Life Measurement and Current Consumption technique guide [13] provides some fundamental information about the procedure. Following and tailoring the guide as required, we executed the following steps to measure current drain of the mobile phone and to model the overall battery consumption:

1) Mobile phone battery was replaced with a 3D-printed dummy battery fixture.
2) The dummy battery was connected to the DC power supply device capable of meeting the minimum measurement requirements specified in the guide.
3) The DC power source was configured to maintain a 3.8V across the dummy battery terminal.
4) The mobile phone was activated. First the mobile phone was initialized back to factory settings, only necessary network settings configured and unrelated system services disabled or not configured. At the beginning of each test run the mobile phone was restarted.
5) We waited 3 minutes for the mobile phone boot process to be completed
6) The mobile phone placed into appropriate test configuration and we waited at least one minute before recording any sample. That waiting time ensured that the mobile phone display went to dim and any application's initial CPU and network activity completed. For each case we stripped down the IMS/RCS application to isolate and model the battery usage of each feature –registration, presence on and IMAP query– and developed several small clients each representing only one of the atomic SIP or IMAP protocol operations. This approach proved to be more practical and cleaner from a software development perspective.
7) Samples were recorded for at least 20 minutes. The application was configured to repeat the predefined network data transfer (predefined amount of atomic SIP or IMAP protocol operations) for 10 minutes and then stopped. We continued to record the samples for another 10 minutes to measure the amount of current being drained when the mobile phone is in IDLE state. We observed that recording the samples for application network activity and IDLE state in different test runs provided biased results due to the rapid changes in the network environment during the day. Recording the samples for both cases in a continuous single test provided more accurate and dependable results since the amount of current required for each operation was calculated by taking the difference of average current drained during these two periods.
8) Average current drain obtained from the measured samples and the amount of current required for each atomic operation was calculated.
9) We repeated each test three times and took their average [5].
10) We modeled the overall battery consumption of the application given specific timer settings being able to set the frequency and the count of these atomic operations in each feature.
11) With these findings we then obtained the optimum timer settings for each feature in order to decrease IMS/RCS application contribution to the battery consumption to as low as possible while still providing a good user experience. In addition to that we discovered the effect of various timer settings on the battery life time in comparison to the mobile phone factory IDLE state.

We executed the tests for mobile phone connected to both a dedicated and shared Wi-Fi access point. Figure 3 depicts the results in both cases. To highlight the difference in the graphics, we set the timer settings to the magnitude of seconds instead of minutes.

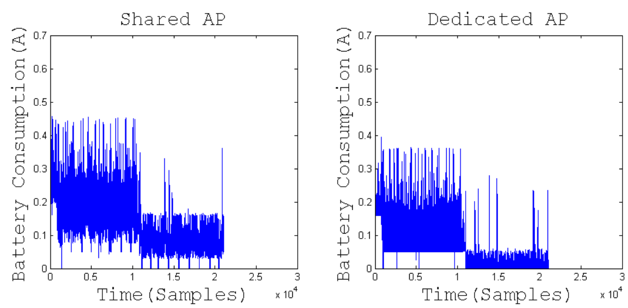

Fig. 3. Amount of current drained connecting to a shared(left) vs a dedicated(right) Wi-Fi access point

In the shared Wi-Fi access point case, the network was heavily loaded which resulted in the amount of current being drained in both cases (network data transfer and IDLE state) higher than the amount of current being drained when connected to a dedicated Wi-Fi access point. However, since we were mainly interested in the additional amount of energy needed for executing a specific task, we found that in both cases our measurements and calculations provided very close results. In other words, no matter which access point the mobile phone was connected to, the additional amount of energy required for a SIP or IMAP operation was very close. Having that information, we continued our tests using a dedicated access point since it enabled us to regenerate the test cases again.

V. Test Results And Discussions

The detailed measurement results enabled us to find out the additional cost of specific atomic operations. Having this information we then obtained the total amount of information required to carry out a feature that consists of multiple atomic operations. And then we calculated the contribution of the always-on features of the application to the total battery consumption and the ratio of contribution for each feature as well.

These findings enabled us to simulate different scenarios, prioritize the features and find their optimum largest values while still providing a decent user experience as well as estimating the battery lifetime for a requested timer configuration. Figure 4 depicts a simulation for given sample timer settings. Note that these timer settings are selected to be in the magnitude of minutes.

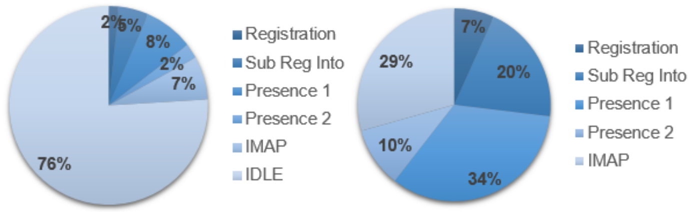

Fig. 4. IMS/RCS application's always-on power consumption compared to phone IDLE state power consumption (left). Contribution of each feature of the IMS/RCS application to overall always-on battery consumption (right)

The drawback of this modelling was isolating all features which is not always the case in real life scenarios. In our test case we had two possible parallel connection approaches. 1) Network data transfer during an active 3G voice call or a separate TCP downloads. 2) With clever timer alignment and intensional delays, grouping the operations and performing them in the same time which would result less amount of energy being consumed per operation. For the first approach, Nurminen [5] showed that parallel connections have significant impact on the battery consumption. The second approach is what we recommended to the application developers to improve the battery performance of the application.

Utilizing the GSM and 3G radio interfaces for the always-on feature network data transfer would obviously provide different results for the exactly same tasks due to the amount of the tail energy being different. In section III we have discussed the differences between the various radio interfaces in terms of tail energy. In our research we have avoided 3G and GSM interfaces due to priority of the Wi-Fi use cases and due to the shortcomings of the modelling attempts for the 3G and GSM interfaces [11].

Some of the lessons we learned in this research:

- GSM Association Battery Life Measurement and Current Consumption Technique guide with some tailoring provided reasonable fundamental knowledge.
- Recording the drained current samples for both data transfer and IDLE state in a continuous single test provided more accurate and dependable results.
- Stripping down the application to isolate the features and model the battery usage of each feature within several small clients each representing only one of the atomic operations helped to build the model.
- With some constraints, modelling indeed did help to estimate the mobile phone battery lifetime and to improve the battery performance of an application.
- When the mobile phone was connected to a Wi-Fi access point which was already heavily used for network data transfer by the other network nodes, the amount of current being drained by the mobile phone itself was much more due to higher maintenance overhead energy for keeping the Wi-Fi interface on.

VI. Conclusion

Within this research we presented a practical methodology for modelling and improving the mobile phone battery consumption of an always-on application. The IMS/RCS application we analyzed periodically updates and retrieves the availability status as well as inbox query task to/from the server over Wi-Fi interface. Those periodical updates and retrievals consist of several atomic operations that implement the various always-on features.

Our approach helped to determine the amount of energy consumed for each always-on feature and model the overall battery consumption of the application and find the optimum timer settings for each feature to minimize the applications' contribution to the battery consumption. Since we used the existing tools with additional in-house application development know-how, we believe this approach can be easily adopted for modelling and improving the battery performance of similar applications and the methodology we presented here can be used as a guideline for measuring the battery performance of diverse mobile phone applications in general.

References

[1] SummitTech, "Rich communication suite," http://www.richcommunicationsuite.com/solution.htm, 2014.

[2] V. Kononen and P. Paakkonen, "Optimizing power consumption of always-on applications based on timer alignment," in *Communication Systems and Networks (COMSNETS), 2011 Third International Conference on*, January 2011.

[3] X. Chen, Y. Chen, Z. Ma, and F. C. A. Fernandes, "How is energy consumed in smartphone display applications?" in *Proceedings of the 14th Workshop on Mobile Computing Systems and Applications*, ser. HotMobile '13. New York, NY, USA: ACM, 2013.

[4] I. Constandache, S. Gaonkar, M. Sayler, R. Choudhury, and L. Cox, "Enloc: Energy-efficient localization for mobile phones," in *INFOCOM 2009, IEEE*, April 2009.

[5] J. Nurminen, "Parallel connections and their effect on the battery consumption of a mobile phone," in *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, January 2010.

[6] G. Perrucci, F. Fitzek, G. Sasso, W. Kellerer, and J. Widmer, "On the impact of 2g and 3g network usage for mobile phones' battery life," in *Wireless Conference, 2009. EW 2009. European*, May 2009.

[7] H. Haverinen, J. Siren, and P. Eronen, "Energy consumption of always-on applications in wcdma networks," in *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, April 2007.

[8] A. Schulman, V. Navda, R. Ramjee, N. Spring, P. Deshp, C. Grunewald, K. Jain, and V. N. Padmanabhan, "Bartendr: A practical approach to energy-aware cellular data scheduling," in *In Mobicom*, 2010.

[9] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. P. Dick, Z. M. Mao, and L. Yang, "Accurate online power estimation and automatic battery behavior based power model generation for smartphones," in *Proceedings of the Eighth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis*, ser. CODES/ISSS '10. New York, NY, USA: ACM, 2010.

[10] R. Murmuria, J. Medsger, A. Stavrou, and J. M. Voas, "Mobile application and device power usage measurements 1,2," 2012.

[11] J. Bornholt, T. Mytkowicz, and K. S. McKinley, "The model is not enough-understanding energy consumption in mobile devices." Hot Chips, August 2012. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=202172

[12] N. Balasubramanian, A. Balasubramanian, and A. 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*, ser. IMC '09. New York, NY, USA: ACM, 2009.

[13] G. Association, "Battery life measurement and current consumption technique," June 2013.

[14] Agilent, "Agilent 6632b high performance dc power supply," http://www.keysight.com/en/pd-838596-pn-6632B, 2014.