Communication Energy Overhead of Mobiles Games

Mathias Almquist, Viktor Almquist, Ekhiotz Jon Vergara, Simin Nadjm-Tehrani
Department of Computer and Information Science
Linköping University, Sweden
matal092@student.liu.se, vikal046@student.liu.se,
ekhiotz.vergara@liu.se, simin.nadjm-tehrani@liu.se

ABSTRACT

Although a significant proportion of the mobile apps are games there has been little attention paid to their specific characteristics with respect to communication energy. In this paper we select 20 mobile games among the top 100 free Android games, and study their data patterns and communication energy use over a total of 25 hours of playing. The analysis of the energy for communication over 3G networks indicates that there is a wide variation among the games, the largest footprint being 8 times higher than the lowest one. The results also indicates both app-specific and category-specific relations between data pattern and energy use, as well as variations in CPU utilisation.

Categories and Subject Descriptors

C.2.1 [**Computer Communication Networks**]: Wireless communication; C.4 [**Performance of Systems**]: Measurement techniques

General Terms

Design, Measurement

Keywords

mobile games; communication energy; 3G; Android;

1. INTRODUCTION

Similar to other application areas, the ubiquity of smartphones has affected the characteristics of the gaming practices tremendously. Mobile players have now access to thousands of games at their fingertips and this has created the opportunity for many game developers to enter the market at an accelerated rate, as well as additional services (e.g., ad sales) to emerge as part of some offerings. In July 2014 Google announced that Android has over 1 billion active monthly users[1] and Google Play Store offers 1.5 million applications where around 18% of all applications are games[2].

However, as we exemplify in this paper, the communication energy landscape of the smartphone usage is still underdeveloped and we see the reduced availability as a result of battery drain a relevant topic with potential in research and development. This is primarily due to the ecosystem of the mobile applications where the resource availability issues need to be addressed at several points: the user end, the data transmission infrastructure, and the cloud/service end, all influenced by hardware characteristics and protocol/software designs.

Earlier studies have shown that when it comes to energy as a resource, the end user efficiency is dependent on device hardware, a complex interaction between the platform services (e.g., Android services), communication provider (telecom operator), and the thousands of apps that a user may download on the device, as well as interactions between these. It is also clear that cellular data communication energy footprint is an element worth a closer look, to understand and optimise [7,15]. A previous study has shown that it can contribute to more than 50% of the total energy consumed by a mobile application (e.g., Angry Birds) over 3G communication [9]. In this paper we focus on the 3G energy footprint of a selection of free Android mobile games, and characterise their comparative energy use. Our study shows that there is a big gap in terms of energy use between two games that have the same gameplay, and that background traffic not relevant to the gameplay can be a major source energy waste.

The contributions of the paper are as follows: (a) We select 20 mobile games amongst the top 100 Android official market free games and create a substantial data set consisting of actual data communication traces while playing the games over a total of 25 hours, (b) we quantify the 3G communication energy of all these games, analyse their contributing components, and capture CPU utilisation of the device while the games are being played.

The paper is structured as follows: Section 2 provides a background on the characteristics of the 3G communication energy and presents the related works. Section 3 describes the game selection methodology laying out the interesting criteria for the comparative study. Section 4 describes our evaluation setup and metrics, followed by section 5 in which

MobiGames'15, May 19 - May 19 2015, Florence, Italy.
ACM 978-1-4503-3499-0/15/05.
http://dx.doi.org/10.1145/2751496.2751498.

[1] http://www.theverge.com/2014/6/25/5841924/google-android-users-1-billion-stats

[2] http://www.appbrain.com/stats/android-market-app-categories, accessed 26 February 2015

individual and comparative analysis of the 20 games with respect to communication energy is carried out. Section 6 shows CPU utilisation for the same set of games, and section 7 concludes the paper.

2. BACKGROUND AND RELATED WORKS

This section begins by providing a background on the energy consumption due to communication for the third generation Universal Mobile Telecommunications System (UMTS) also known as 3G at the user equipment (UE) side and presents the related works.

2.1 Energy footprint of 3G

The energy consumption of the UE when connected to a 3G network is mostly determined by the radio resource management performed at the network operator end using the Radio Resource Control (RRC) and Radio Link Control (RLC) protocols.

According to the RRC, the UE implements a state machine where each state has different power consumption and performance in terms of data rate and latency. The states are Dedicated Channel or CELL_DCH (DCH), Forward Access Channel or CELL_FACH (FACH), and Paging Channel[3] or CELL_PCH and URA_PCH (PCH) sorted by power consumption and performance in descending order. Fig. 1 shows an example of the power consumption for the state machine implemented by the operator TeliaSonera in Sweden. The bottom graph shows the data sent.

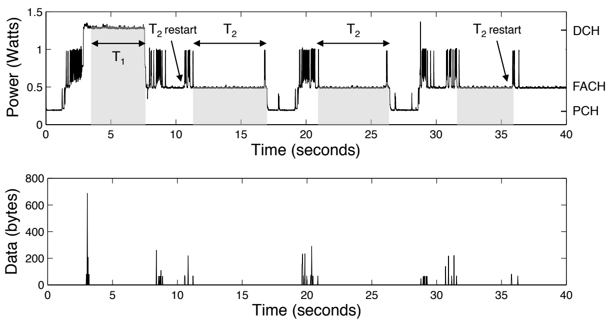

Figure 1: Example power profile for 3G using a mobile broadband module (Ericsson F3307).

The UE does not transmit any data in PCH, so when the UE generates or receives traffic some signalling is performed to move the UE to either FACH or DCH. The UE reports the observed traffic volume using the RLC protocol which is used to evaluate the resource allocation and control state transitions to higher performance states. A transition is triggered when the RLC buffer data occupancy of the UE exceeds some fixed RLC buffer threshold in the uplink or downlink. These thresholds control the PCH-DCH, PCH-FACH and FACH-DCH state transitions. The signalling occurs before data reception in Fig. 1 at 3 s since a downlink packet is received.

The UE is allocated a dedicated physical channel (uplink and downlink) in DCH providing the highest data rates. The RRC state machine uses inactivity timers to down switch the UE to lower performance states. The UE is moved to FACH after T_1 when there is small or no data transmission. The UE experiences lower data rates in FACH. Finally, the inactivity timer T_2 controls the FACH to PCH transition.

Inactivity timers create energy overheads known as *energy tails* since the UE remains in a high energy consuming state while not transmitting anything. Fig. 1 is a good illustration showing that the data pattern clearly impacts the energy consumption in a complex way.

2.2 Related works

This section provides the related work divided into two main topics: energy consumption due to communication, and analysis of mobile games.

Mobile communication energy: Several works [7, 9, 12, 15] present tools to support the developers in analysing energy consumption of applications.

Optimising energy consumption often includes adopting traffic shaping methods such as batching or prefetching data [2], or exploiting radio resource control techniques [1]. For example, our previous work [14] shows that even the short instant messaging sessions over 3G consume high energy and proposes message batching strategies based on application level indicators.

Mobile games: Mobile games have not been subject to extensive attention in terms of energy studies. ARIVU [3] is an energy-aware middleware designed to reduce the energy consumption of the WiFi interface when playing online multiplayer games. Prokova et al. [11] compare the average power usage of Android games with and without ads connected via WiFi. Our work focuses on the 3G energy consumption and shows that game-specific background data is a major contributor in addition to ad-related traffic.

Patro et al. [10] study data collected from a massively multiplayer game over a three year period and show, among other results, that around 70% of the sessions are 20 minutes or shorter. Another study of a single player game by Obermiller and Bayless [8] points out that half of the players played the game only a few times and for 15 minutes. We use this earlier data to justify that our selection of the duration of each play is reasonable given that we had to perform each session manually.

3. MOBILE GAME SELECTION

A core aspect of our study is the selection of a set of games to be profiled, which need to be representative in terms of popularity, type and characteristics. The selection of games needs to provide diversity to support a study with a wide range of communication patterns and their impact on energy consumption. We consider the following game properties:

Gameplay: While it is natural to select games with different gameplay to provide diversity, we also select games with similar gameplay in order to compare them. We consider simple and advanced games in terms of graphics that can impact the CPU usage.

Multiplayer game: Online multiplayer games are expected to generate more traffic than single player games since they communicate with other players and/or servers.

Input type: We categorise the input type of mobile games based on our previous analysis [6]: The state of a game with a *time-insensitive* input changes only when an input is performed by the user (e.g., Tic-Tac-Toe). A *time-sensitive* input is required in games where the state can change without

[3] Since CELL_PCH and URA_PCH result in similar energy consumption we consider them as PCH for simplicity.

user interaction, e.g., driving games or controlling a character's movement.

Advertisements: We limit this work to free games and therefore we also study whether the games contain advertisements with an impact on the amount of data sent and energy consumption.

Release date: Selecting current games as well as older games allows us to observe potential trends.

The games are selected from the top 100 free games from the Google Play Store on 26th May 2014 based on (1) the above described properties, and (2) a subjective coverage of the known game types based on authors' gaming experience. Selecting games based on the above properties provides us the possibility of comparing aspects such as sequels and predecessors, single player and online multiplayer games, current and earlier games or games with different gameplay.

Tables 1, 2 and 3 show the games selected categorised by single player time-sensitive and time-insensitive games and multiplayer online games.

4. EVALUATION SETUP AND METRICS

In order to analyse the energy consumption due to communication and the CPU usage at the user end we employ a Samsung I9300 Galaxy S3 running Android 4.1.2, where we install the game under test. Since our aim is to quantify the energy consumption due to realistic playing sessions, for our experiments a human player interacts with the game. We treat the games as a black box since their source code is not available.

Trace collection: We collect the packet traces while playing the game under test using *tcpdump*. These are later used to compute the energy consumption. Only the traffic of the game under test is allowed using a *iptables*-based firewall to block the rest of the traffic.

We collect CPU load information for the game under test using *top*, a Unix command line tool that continuously monitors the list of processes running in the system and provides CPU usage, memory usage and runtime for each process. The Android Debug Bridge command line tool is employed in a laptop connected via USB to the mobile device to start the aforementioned tools in the device.

Number of tests: Each of the 20 games is played 5 times for 15 minutes. Since the gaming sessions are quite repetitive, playing longer sessions does not necessarily add more interesting information. We play 75 minutes per game and 25 hours in total.

Communication energy: The energy consumption for each test is computed from the gathered traces using the EnergyBox, a parametrised energy simulation tool that employes RRC state machine parameters, device-specific power levels and packet traces as input. EnergyBox has been evaluated against physical energy consumption measurements showing an average accuracy of 98% [15].

For a given packet trace, EnergyBox derives the UE state performing iterative packet-driven simulation given the input 3G network parameters. The RRC state machine is captured by a finite state machine that simulates state transitions using the state transition times, RLC buffer thresholds and inactivity timers. The energy consumption is then calculated by associating the input UE-specific power levels with the emulated intervals in each state, and integrating them over time.

For our tests the parameters of EnergyBox correspond to the operator TeliaSonera measured in our local experimental area. The inactivity timers are set T_1=4.1 s and T_2=5.6 s respectively. The RLC buffer thresholds correspond to B_1^u = 1000 and B_2^u = 294 bytes for uplink, and B_1^d = 515 and B_2^d = 515 bytes for downlink. The state transition time due to signalling is set to 1.7 s, 0.65 s and 0.435 s for PCH-DCH, FACH-DCH and PCH-FACH respectively. The UE power values for the RRC states are based on earlier measurements [14]: DCH = 612 mW, FACH = 416 mW. We set PCH = 0 W in order to quantify only the energy spent for data communication since the measurement period is the same for all games.

Evaluation metrics: The metrics used to analyse the collected data are the amount of data sent and total energy consumption. These are interesting from the user perspective since they directly impact user's battery lifetime and data plan quota.

We also analyse the time spend by the UE in the different RRC states as well as the number and type of RRC state transitions which provide valuable insight regarding the impact of the data pattern and the RRC state machine. The number of RRC state transitions is a representative metric correlating with the amount of signalling created between the UE and the network operator. For example, a single FACH-DCH state transition is associated with 10 message signalling transmissions [13], which introduces additional delay and energy consumption at the user end and overhead for the network operator end.

Finally, we analyse the average CPU load of the games during the playing session.

5. 3G ENERGY OF GAMES

This section presents the results regarding 3G communication energy, amount of data sent and RRC state machine statistics. We also provide some general guidelines to improve games' energy consumption.

5.1 Energy consumption and data traffic

Fig. 2 shows the average energy consumption due to communication as well as the standard deviation of the 5 repetitions per game. The games are grouped by the categories as presented in Tables 1, 2 and 3.

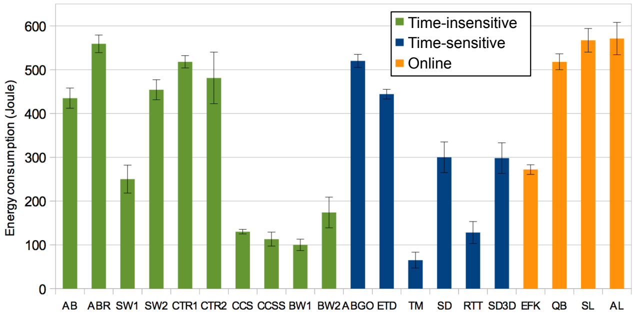

Figure 2: Average energy consumption for the selected games due to communication.

There is an evident diversity in energy consumption among the 20 games. The most and least consuming games are Arcane Legends (AL) and Tower Madness (TM) consuming

Table 1: Single player games with time-insensitive input.

ID	Game	Gameplay	Ads	Release date	Version
AB	Angry Birds	A 2D game where birds are fired at enemies from a slingshot.	✓	Dec. 2009	2.2.0
ABR	Angry Birds RIO	A newer version of Angry Birds with similar gameplay.	✓	Mar. 2011	2.2.0
SW1	Angry Birds Star Wars 1	Another version of Angry Birds with Star Wars theme.	✓	Nov. 2012	1.5.2
SW2	Angry Birds Star Wars 2	A newer version of Angry Birds Star Wars.	✓	Sep. 2013	1.6.0
CTR1	Cut the Rope 1	2D physics-based puzzle game. The player cuts ropes to feed candy to a monster.	✓	Oct. 2010	2.4.2
CTR2	Cut the Rope 2	The newest version of Cut the Rope with the same gameplay	✓	Dec. 2013	1.1.7
CCS	Candy Crush Saga	A match-three puzzle game where you group up same coloured candy to complete levels.	x	Nov. 2012	1.34.1
CCSS	Candy Crush Soda Saga	A newer version of Candy Crush Saga with some new features.	x	Jul. 2012	1.25.21
BW1	Bubble Witch Saga 1	Classic match-three game. The player aims and shoots bubbles at other bubbles of the same colour to destroy them.	x	Jul. 2012	3.1.8
BW2	Bubble Witch Saga 2	The newest version of Bubble Witch Saga.	x	Jun. 2014	1.4.2

Table 2: Single player games with time-sensitive input.

ID	Game	Gameplay	Ads	Release date	Version
ABGO	Angry Birds GO!	3D kart-racing game.	✓	Dec. 2013	1.4.3
ETD	Elemental Tower Defense	Tower defense game where the player builds towers to shoot at enemies to stop them from reaching a point in the map.	✓	May 2014	1.0.57
TM	Tower Madness Zero	Tower defense game with similar gameplay.	✓	Oct. 2009	1.21
SD	Sonic Dash	3D endless platform runner game where the player collects items and avoids obstacles.	✓	Mar. 2013	1.14.0.Go
RTT	Race the Traffic	3D driving game where the player avoids crashing.	✓	Jul. 2014	1.0.6
SD3D	School Driving 3D	3D driving game. The player drives in a city learning how to drive.	✓	Apr. 2014	1.7.0

Table 3: Multiplayer online games.

ID	Game	Gameplay	Ads	Release date	Version
EFK	Empire four Kingdoms	Strategic game where the player builds a castle, creates an army and fights other players	x	Mar. 2013	1.10.60
QB	Quiz Battle	Turn-based quiz game.	x	Aug. 2012	1.3.102
SL	Star Legends	Multiplayer role playing game. Fight monsters with other players.	x	Q2 2011	2.0.1.0
AL	Arcane Legends	Multiplayer role playing game which is a sequel of Star Legends.	x	Nov. 2012	1.1.0.1

561 and 65 Joules respectively. The results show that half of the games present a high energy consumption (more than 435 Joules).

Multiplayer online games (AL, SL and QB) are among the highest consumers as expected. Surprisingly, single player games like Angry Birds (AB, ABR, SW2 and ABGO) and Cut the Rope games (CTR1 and CTR2) consume as much as online multiplayer games.

Fig. 3 shows the average data traffic and standard deviation for the 20 games. The average energy consumption is shown in the x-axis for each of the games for comparison purposes.

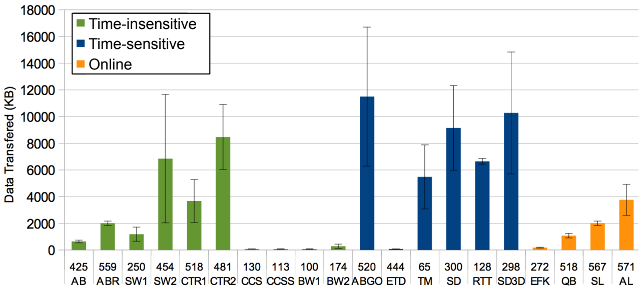

Figure 3: Average data traffic by the 20 games.

Similarly to energy consumption, there is a big diversity in the amount of data traffic. It is noticeable that the energy consumption is not proportional to the data sent by simply comparing Elemental Tower Defense (ETD) and Tower Madness (TM). ETD consumes 7 times more energy while TM sends approximately 63 times more data.

The difference in energy and amount of data sent by the single player games with and without in-game advertisements is enormous. Only 4 out of the 16 single player games are ad-free (CSS, CCSS, BW1 and BW2) and are developed by the same company. These ad-free games present low energy consumption as well as low amount of data sent. These games serve as an illustrative example of the consumption of games without any ads.

We also observe a large standard deviation in Fig. 3 for the games with the largest amount of data sent, such as Angry Birds RIO (ABR). The number of ads fetched and statistics sent is different for each game and the ads come in different formats (e.g., video or images), which results in a large deviation.

5.2 UE states and data pattern

Clearly, a game should make the UE spend the largest percentage of time in the PCH state to reduce its consumption and avoid staying long in FACH and DCH. Fig. 4 shows that the RRC state distribution differs greatly which indicates different data patterns. We show the RRC state distribution for the trace with the median energy for each game.

The most consuming games (ABR, SL and AL) spend approximately 95% of the time in DCH. Games with similar gameplay such as Angry Birds (SW1, SW2 and ABR) or Sonic Dash, School Driving 3D and Angry Birds GO (SD, SD3D and ABGO) present very different RRC state distributions, showing that there is room to improve the networking behaviour of these games.

A more detailed analysis of the individual packet traces reveals that most transmissions are used for fetching ads, sending game statistics such as achievements, score or items

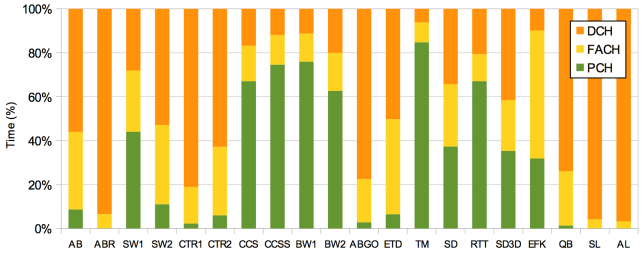

Figure 4: Distribution of time spent by the UE in the different RRC states for each game.

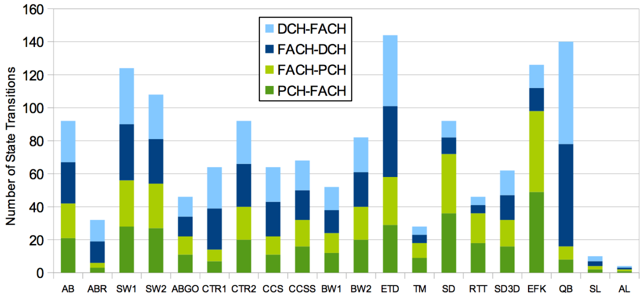

Figure 5: Number of RRC state transitions for the 20 games.

purchased, and encrypted data which we cannot access. For example, the single player games without ads (CSS, CCSS, BW1 and BW2) communicate with the company server's (King) as well as cloud computing service providers (Amazon). The game statistics sent by Bubble Witch 2 are accessible in plain text.

The multiplayer game Empire Four Kingdoms (EFK) spends a significant percentage of the time in PCH and FACH reducing significantly the time in DCH. Even though the game sends packets often, these are small enough to not exceed the RLC buffer threshold and avoid expensive transitions to DCH.

We look in more detail at Tower Madness (TM), since it is the lowest consuming game even though it employes in-game advertisements, and compare it against ETD. We observe that ETD transmits data constantly and fetches ads frequently (roughly every 20 s) spending most of the time in FACH and DCH. Instead, TM fetches ads only when a new level is entered (roughly 5-10 mins) staying most of the time in PCH. However, as previously described, TM sends much more data since it receives video ads in large files while ETD receives small images. TM is a good example of how to keep the energy consumption low while still using ads.

An interesting observation from Fig. 4 is that out of the single player games with in-game ads, the majority of games with time-sensitive input (TM, SD, RTT and SD3D) spend more time in PCH than the ones with time-insensitive input (AB, ABR, SW2, CTR1 and CTR2). The majority of games fetch ads when entering, restarting or completing levels. We observe that completing a level in the selected time-insensitive games takes short time (approximately 40 s) resulting in more frequent downloads than for time-sensitive games which take around 2 to 10 mins to complete a level. Thus, the time to complete a level seems to impact the energy consumption through the ad-fetching frequency.

Fig. 5 shows the number of state transitions performed by the UE when running each game. We show the run with the median energy consumption.

There is a noticeable difference among the multiplayer games, where Star Legends (SL) and Arcane Legends (AL) are real-time games requiring frequent updates compared to the other games (QB and EFK).

Some games (SD, RTT and EFK) employ small data transmissions resulting in a majority of PCH-FACH transitions. This appears as a favourable behaviour which avoids transitions to expensive DCH state allowing them to send data at lower energy cost.

Surprisingly, we show that all games except the multiplayer online games can be played without connectivity. 16 out of 20 games can still be played if we enable firewall to block the network traffic of the games, which makes the high energy consumption unjustifiable from the user perspective and points out the need to reduce their energy consumption.

To sum up, our results show that there is room for improvement and that the energy consumption can be reduced if we focus on communication.

5.3 Suggestions for improvement

This section provides some general guidelines to reduce the energy consumption of mobile games from the application developer perspective.

Energy efficiency starts from energy awareness, and thus we strongly recommend that developers integrate the freely available energy simulation tools such as EnergyBox within their testing environment [7, 9, 12, 15]. These tools can help to reveal inefficiencies in a similar way to our study.

Even though the amount of data sent is not proportional to energy consumption, reducing the number of transmissions over time is a good starting point, which can easily extend the idle time of the UE. Thus, bursty communication should be preferred to continuous data transmission.

Different techniques have already been studied to improve the energy efficiency of communication by adapting the communication pattern, such as batching, piggybacking and data prefetching [2, 14]. Even though the cellular interface is shared among all the apps and the developer cannot influence the rest of the apps to coordinate and batch transmissions, the app developers can leverage the coordinated batching and piggybacking techniques provided by the operating system (e.g., the background JobScheduler API of Project Volta for Android [4]). Hence, the game content like statistics could be categorised as background data and deferred for a more energy-efficient operation.

We showed that downloading ads inefficiently can be very expensive and that the majority of the games download ads periodically or when entering, restarting or completing a level. While showing ads is a foreground task which cannot be deferred, prefetching can be employed to download several ads at once to decrease the fetching frequency. This can reduce the energy consumption of games with short level completion times or short ad update periods. For video ads, downloading them at once at the highest data rate is recommended, since the user often needs to watch the whole video before starting the game.

Finally, data compression is a double-edge technique, which can sometimes benefit [16], but it is not always efficient at the application level if the data sent is small [14].

6. CPU LOAD

Fig. 6 shows the average CPU utilisation and standard deviation for each game. The results show that the CPU utilisation of the games is between 14-51% presenting a lower diversity. The standard deviation for most games is quite low indicating a similar CPU usage per round.

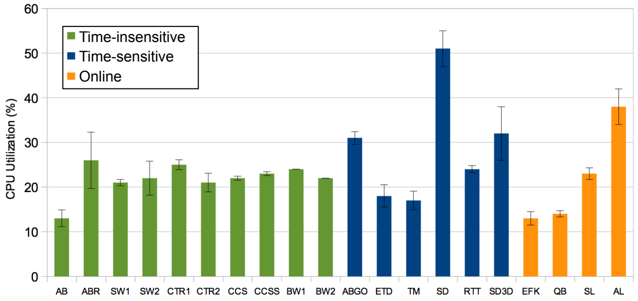

Figure 6: Average CPU load of the 20 games.

We observe that highest load is created by games which employ 3D graphics (ABGO, SD, SD3D and AL). Other 3D games (SL and RTT) present also a considerable CPU load. Sonic Dash (SD) generates the highest CPU load, which is expected since it has a fast-paced gameplay and advanced graphics.

It is noticeable that similar games have quite different CPU load (e.g., AB and ABR). Ma et al. [5] show that a higher CPU utilisation drains the battery much quicker, which given our results indicates that there is also room for code optimisation.

7. CONCLUSION AND FUTURE WORK

To the best of our knowledge this paper is the first attempt at creating a large enough dataset of free mobiles games played by human players for investigating the data patterns and cellular communication energy consumption properties. This dataset is available to the research community for further analysis.

As part of our initial findings the dataset confirms the fact that games (like other mobile apps) do not have an energy-proportional communication over 3G. The analysis revealed a wide variation in consumption among the selected top free games, but also confirmed the expectation that ad-free games have much lower data traffic and lower energy consumption in general. There is evidence that it is feasible to combine low energy consumption and advertisements. Also, it is currently possible to play the games by disabling network connections during playing to avoid the additional communication energy.

There are many interesting directions to continue this work, among them evaluation of batching techniques for various categories of background traffic (including ads, game statistics, app-specific data). Performing any such batching is harder for online games, and studying the interplay between the potentially added delays and the behaviour of online game is an interesting topic for future work. Finally, analysing the contribution to the total energy by other aspects than communication is an interesting direction.

Acknowledgements

This work was supported by the Swedish national graduate school in computer science (CUGS).

8. REFERENCES

[1] P. K. Athivarapu, R. Bhagwan, S. Guha, V. Navda, R. Ramjee, D. Arora, V. N. Padmanabhan, and G. Varghese. Radiojockey: mining program execution to optimize cellular radio usage. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, pages 101–112. ACM, 2012.

[2] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani. Energy consumption in mobile phones: a measurement study and implications for network applications. In *IMC'09*, pages 280–293. ACM, 2009.

[3] A. Bhojan, A. L. Akhihebbal, M. C. Chan, and R. K. Balan. Arivu: Making networked mobile games green. *Mob. Netw. Appl.*, 17(1):21–28, Feb. 2012.

[4] Google. Project Volta. Accessed 31st March 2015. https://developer.android.com/about/versions/android-5.0.html#power.

[5] X. Ma, Z. Deng, M. Dong, and L. Zhong. Characterizing the Performance and Power Consumption of 3D Mobile Games. *Computer*, 46(4):76–82, 2013.

[6] J. Marian Santos, S. Nadjm-Tehrani, and A. Bianzino. Monkey Gamer: Automatic profiling of Android games. In *6th International Conference on Mobile Computing, Applications and Services (MobiCASE)*, pages 125–132. IEEE, Nov 2014.

[7] R. Mittal, A. Kansal, and R. Chandra. Empowering Developers to Estimate App Energy Consumption. MobiCom '12, pages 317–328. ACM, 2012.

[8] I. Obermiller and S. Bayless. Legends of Descent: Analytics in an Ad-supported Windows Phone Game. In *Proceedings of the First ACM International Workshop on Mobile Gaming*, MobileGames '12, pages 1–6. ACM, 2012.

[9] A. Pathak, Y. C. Hu, and M. 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*, EuroSys '12, pages 29–42, 2012.

[10] A. Patro, S. Rayanchu, M. Griepentrog, Y. Ma, and S. Banerjee. The Anatomy of a Large Mobile Massively Multiplayer Online Game. *SIGCOMM Comput. Commun. Rev.*, 42(4):479–484, Sept. 2012.

[11] I. Prochkova, V. Singh, and J. K. Nurminen. Energy Cost of Advertisements in Mobile Games on the Android Platform. In *Proceedings of the Sixth International Conference on Next Generation Mobile Applications, Services and Technologies*, NGMAST '12, pages 147–152. IEEE, 2012.

[12] F. Qian, Z. Wang, A. Gerber, Z. Mao, S. Sen, and O. Spatscheck. Profiling resource usage for mobile applications: A cross-layer approach. MobiSys '11, pages 321–334. ACM, 2011.

[13] L. Qian, E. W. W. Chan, P. P. C. Lee, and C. He. Characterization of 3G Control-Plane Signaling Overhead from a Data-Plane Perspective. MSWiM '12. ACM, 2012.

[14] E. J. Vergara, S. Andersson, and S. Nadjm-Tehrani. When mice consume like elephants: Instant messaging applications. e-Energy '14, pages 97–107. ACM, 2014.

[15] E. J. Vergara, S. Nadjm-Tehrani, and M. Prihodko. Energybox: Disclosing the wireless transmission energy cost for mobile devices. *Sustainable Computing: Informatics and Systems*, 4(2):118 – 135, 2014.

[16] L. Wang and J. Manner. Evaluation of data compression for energy-aware communication in mobile networks. In *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, CyberC '09, pages 69 –76. IEEE, Oct. 2009.