Research Paper on Enhanced Battery for Android Phones using the Power of Cloud through Data Synchronization

[I]Sumandeep Kaur, [II]Sugandha Sharma, [III]Mayank Arora
[I,II]Dept. of CSE, Chandigarh University, Gharuan Mohali, Punjab, India
[III]Dept. of CSE, CCET, Panjab University, Chandigarh, India

Abstract

This article shows a review of the Cloud Computing, mobile Cloud Computing and data synchronization between mobile and cloud. In this paper, the focus has been made on improving the synchronization of smartphones by using Cloud computing and virtualization techniques to shift the workload from merely a smartphone to a resource rich computational Cloud environment. The data will be automatically synchronized to the user's devices when they are connected to internet. This paper provides the user a synchronization mechanism which synchronizes the data automatically between mobile devices and cloud and is energy aware and tries to minimize the energy usage

Keywords

Cloud computing, Mobile cloud computing, Offloading, Data synchronization

I. Introduction

Cloud Computing is a model for enabling convenient, on-demand network services to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [1].

Service Models

IaaS: In Infrastructure as a Service (IaaS), storage, computation, and network resources are the major components that are provided as a service to the customer [2].
PaaS: In the Platform as a Service (PaaS) model this platform provides all the facilities to delivering the applications to end users. Software and service developers are the main users of PaaS.
SaaS: The end user is the customer for Software as a Service (SaaS), since SaaS provides a complete software application running in the cloud[3].

Deployment Models

Public cloud: In the public clouds, multiple users share the computing resources provided by a single service provider.
Private cloud: In private cloud, computing resources are used and controlled by a private enterprise. It is managed by internal employees or service provider [4].
Hybrid cloud: A third type can be hybrid cloud that is typical combination of public and private cloud.
Community cloud:. The cloud infrastructure could be hosted by a third-party dealer or within one of the organizations in the community [5].

Mobile cloud computing

The increasing usage of mobile computing is evident by the study by Juniper Research, which states that the consumer and enterprise market for cloud-based mobile applications is expected to rise to $9.5 billion by 2014. Today, the market of mobile phones is increasing at a very high speed. Everyone has a smart phone which provides the facility to customer move anywhere and access the data anytime or any place. There are some limitations in mobile phones with respect to the desktop these are limited battery life, storage capacity, bandwidth etc. [6].

Advantages of Mobile Cloud Computing-

1) **Extending battery lifetime**- Battery is one of the main concerns for mobile phones. Several solutions have been planned to save the battery life of smartphone and to manage the disk and screen in an intelligent manner to reduce power consumption.
2) **Improving data storage capacity and processing power:** Storage capacity is also a limitation for mobile devices. MCC is developed to support mobile customers to store the large data on the cloud through wireless networks. The example is the Amazon Simple Storage Service (Amazon S3) which supports file storage facility.
3) **Improving reliability:** Storing data or running applications on clouds is an effective way to improve the reliability since the data and application are stored and backed up on a number of computers. This reduces the chance of data and application lost on the mobile phones [7].

Issues in Mobile Cloud Computing

1) **Computing power are limited** -The speed of processors in mobile phones is very slow in comparison with desktop computers. To use high speed processor in mobile devices require more power consumption.
2) **Limited Battery** -Limited battery power of mobile devices is a barrier for interacting via wireless network because it requires high battery consumption [8].

Offloading

Smartphones have become more and more intelligent and powerful. Many complex applications, which used to be only on PCs, have been developed and running on smartphones. However, they also greatly increase the workload on smartphones and introduce a lot of data transmissions overheads between smartphones and telecom networks. The heavy workload and traffic affect both smartphone clients. For users, heavy application or heavy workload and traffic drain smartphone battery quickly [9].

1. **Partial offloading:** Conceptually when an application could be offloaded in parts i.e. only the compute intensive heavy part of the application is offloaded then it is known as Partial Offloading [10].
2. **Complete offloading:** In this offloading method complete application is offloaded to the cloud to save the battery life

of smartphone [11].

II. Proposed Work

As discussed is the previous section it is clear that the synchronization is highly required in case of offloading as the data needs to be consistent in both the Smartphone as well as the Cloud Side failing to which the whole concept of Offloading could fall into jeopardy. Yet keeping the smartphone and the Clone Smartphone on the Cloud side synchronised all the time will lead to loss of energy as sending data over 2G will be very costly in terms of energy. Sudha S et al. proposed architecture to help users to access a large volume of storage on the clone Cloud. They focused on automatic synchronization of data between different devices such as a Mobile and a PC or computer. But again this research has a drawback that a lot of energy would be wasted while keeping the state of the mobile device and the PC running the same application same as the data needs to be physically transferred to the cloud [13]. It is proposed to make the synchronisation process more aware of the user's requirements and priorities as well as the best synchronisation environment.

The above diagram depicts the working of the proposed Architecture. The user gives an input to the smartphone via any medium, it could be an image from the camera or Bluetooth or any other source. The analysis unit starts its work of deciding whether this type of data needs synchronization or not. If synchronisation is required the parameters are collected and passed on to Synchronization Decision system. Depending upon the parameters provided by the Collection unit the decision is made. If the conditions are favourable the synchronization is made at that time itself otherwise the synchronization is made to wait till the timing unit allows it [12].

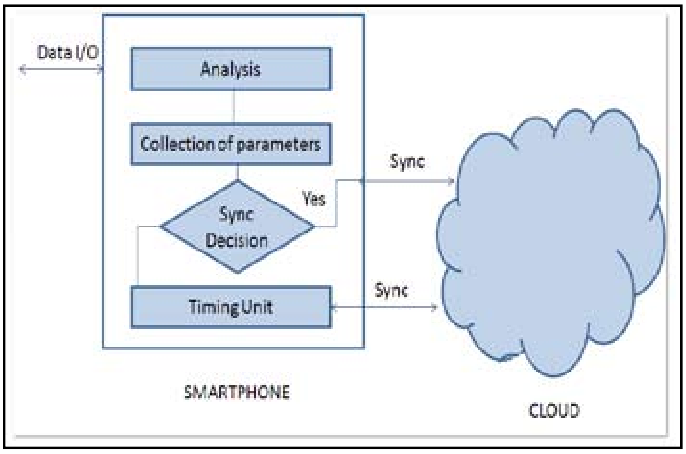

Fig. 1: proposed architecture[12]

Block Diagram

Smartphone side-

a) **I/O Unit-** The input/output unit is responsible for the collection of inputs from the user/web services/sensors.After the execution has been done, this unit provides the output to the user/web services/sensors in a desired format.

b) **Data Analysis-** There could be multiple forms of data required in an application and most of the data need not be synchronised.The data analysis unit will analyse the data for the need of synchronised.

c) **Synchronization Decision Unit**-This unit makes the synchronization decision that is whether synchronization is required for this type of execution or not.The synchronisation decision unit provides its output to the timing unit.

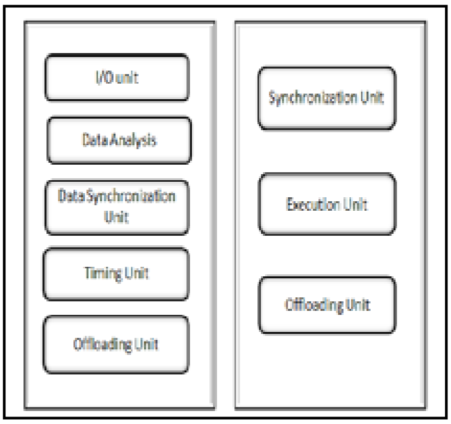

Fig. 2: Block diagram of proposed architecture

d) **Timing Unit**-The timing unit will work as a gatekeeper for the synchronization process.The timing unit will get the information from the synchronization decision unit about when to offload if required. It will delay the offloading till it is actually required.

e) **Offloading Unit**-This unit is responsible for the actual offloading of the data to the cloud[14].

Cloud side

a) **Synchronization Unit**- This unit will receive all the information from the synchronization decision unit on the smartphone side about when the synchronization will take place.

b) **Execution Unit**- The execution unit makes the execution of the code when data is received by the offloading unit. It provides the output to the offloading unit to be sent to the smartphone.

c) **Offloading Unit-** The offloading unit receives the data from the smartphone required by the execution unit to run the computation. The offloading unit is also responsible for sending the data back to the smartphone. The data exchange done by the offloading [17] unit is controlled by the synchronization unit.

III. Results and Disscussion

The above table shows the results of running an compute intensive application i.e. a random number generator on the smartphone without any connection to the cloud. The application running on the smartphone is able to generate random numbers in the range of 100 to 10 Lakhs.

Table 1: Results of running the application on smartphone only.

No.	Total consumption		
	Time taken in msecs.	Energy consumption (in joules)	Battery consumption (in milli watt)
100	2	5.2	2.6
1000	44	5.3	0.1204
10000	338	6.8	0.0201
100000	2972	10.2	0.0034
1000000	42010	46.7	0.0011

Generation of random numbers beyond 10 lakhs is beyond the smartphone's reach. While running the application to create 10 lakh random numbers it took the application more than 42 seconds and around 47 joules of energy [18]was dissipated in this process.

The second table shown here uses an offloading architecture described in [12] to generate the random numbers using the power of cloud. This mechanism uses a complete offloading architecture which offloads the whole application to the cloud.

Table 2: Result of offloading architecture

No.	Total consumption		
	Time taken in msecs.	Energy consumption (in joules)	Battery consumption (in milli watt)
100	25	5.2	0.124
1000	44	5.3	0.0863
10000	73	4.3	0.0589
100000	90	4.5	0.0500
1000000	387	5.7	0.0147
1Cr	1243	8.1	0.0065
10 Cr	9143	13.8	0.0015

The main problem in this architecture is that no synchronization mechanism is described in this mechanism. The idea for this research is to improve the synchronization in the existing architecture and test run it and compare it with the existing architecture. It is also to be noted that as the application is offloaded to the cloud which in our case a type-II Hypervisor is running on a laptop the efficiency increases many folds. The application is now capable of generation over 10 crore random numbers which was not possible in case the same application was running on a smartphone. The application takes around 9 sec to generate 10 Crore
random numbers which was not even possible in case the application was not backed up by the power of cloud.

The third table shows the results of the proposed architecture. The results compared to the smartphone the proposed architecture gives a speedup of almost 50X but if compared to the offloading architecture the results may not be as impressive but battery [16] saving and speedup is achieved. The energy consumed by the offloading architecture while generating 10 crore numbers was 13.8 Joules and the energy consumed by the proposed scheme while creating 10 crores numbers was 10.5 Joules thus saving a significant amount of battery for the smartphone. The results could be more impressive in case the application needs more data processing and hence the application needs to synchronize more and more. The case of image processing applications the results could be even better.

Table 3: Results of running application using proposed architecture

No.	Total consumption		
	Time taken in msecs.	Energy consumption (in joules)	Battery consumption (in milli watt)
100	3	5.2	2.6
1000	44	5.3	0.1204
10000	73	6.8	0.0201
100000	93	10.2	0.0034
1000000	387	46.7	0.0011
1Cr	1243	7.2	0.0054
10 Cr	9143	10.5	0.0012

The following graph shows the comparison of the time taken to generate the random numbers by the the architectures under comparison. As clearly visible the time taken by the standalone smartphone is the greatest and comparatively the time taken by the virtual smartphone architecture is much more than the time taken by the proposed architecture with synchronization.

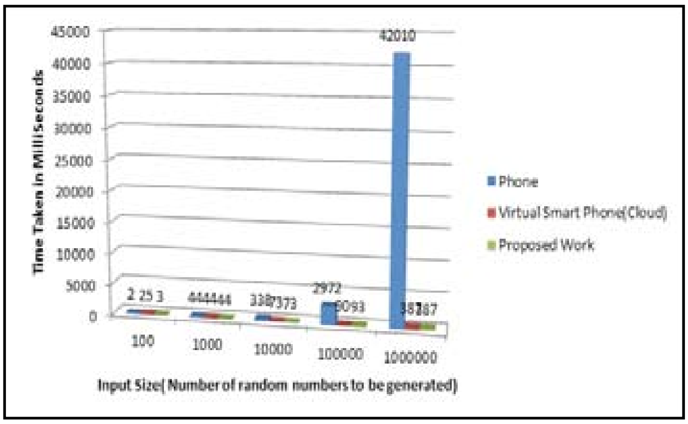

Fig 3: Execution Time Graph Ranging from 100 to 10000.

The graph shown below shows the comparison between the energy consumption by the three architectures. The energy consumed to generate 10 lakh random numbers by the smartphone is about 47 joules whereas the offloading architecture finishes that task in only 5.7 joules. The architecture proposed in this paper does the same work in 4.9 joules which is around 1 joule less than the offloading architecture.

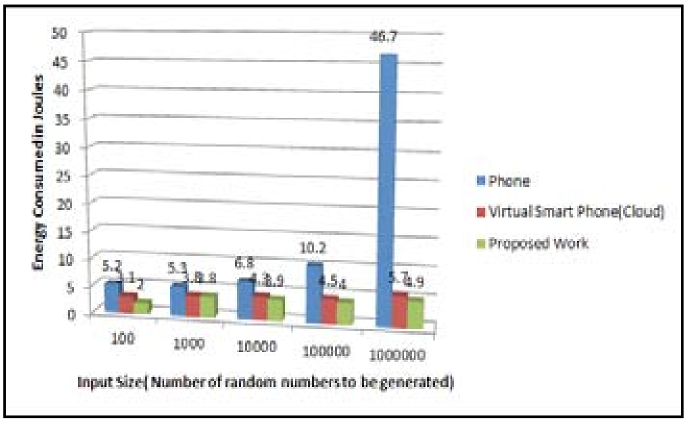

Fig. 4: Energy Consumption Graph

The energy consumption[15] and the time taken by the applications could be further reduced depending upon the application in which the architecture is being used.

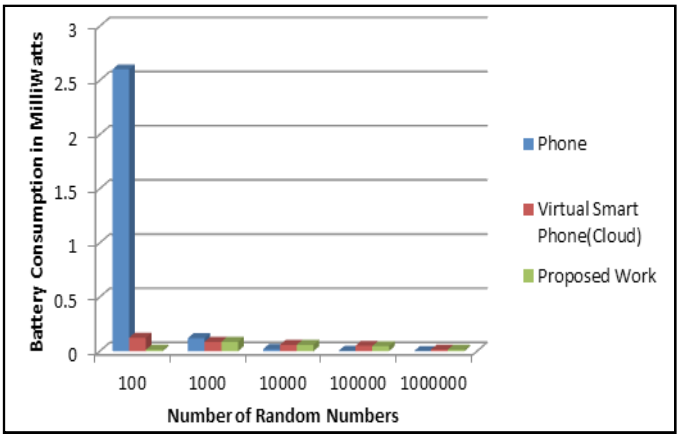

Fig. 5: Battery Consumption Graph

IV. Conclusions

The usage of smart phones is increasing rapidly over the last few years. Due to their mobility, bandwidth and good connectivity, smartphones are increasing thrice as compared to PCs. However they are still limitation by limited processing power, memory and Battery. Offload method reduce the workload of smartphone application. Most of the authors only emphasize on the offloading technique, but only some author emphasize on the synchronization. This paper describe about the synchronization technique to improve the battery consumption.

V. AcknowledgEment

The writing of this dissertation has been assisted by the generous help of many people. The contributions of many different people, in their different ways, have made this possible. I feel that I was very fortunate to receive assistance from them. I would like to extend my appreciation especially to the following.
First and foremost, I am indebted to my supervisor, Sugandha Sharma, Assistant Professor, Dept. of Computer science and engineering, who has been very supportive at every stage of this dissertation. I wish to express my utmost gratitude to her for her invaluable advice and patience in reading, correcting and commenting on the drafts of the dissertation and, more importantly, for her generosity which I has received throughout my entire research program.

References

[1] *The NIST definition of Cloud Computing. Mar 2011. Available:http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc.*

[2] *Alexa Huth and James Cebula "The Basics of Cloud Computing".*

[3] *Xu Wang, Beizhan Wang, Jing Huang"Cloud computing and its key techniques" © 2011 IEEE.*

[4] *Krishan Kant Lavania , Yogita Sharma , Chandresh Bakliwal" A Review on Cloud Computing Model" Volume: 1 Issue: 3, MAR 2013*

[5] *Satish Kumar, Nagesh Salimath"A Survey on Cloud Computing and its Services" Volume : 2 | Issue : 11 | November 2013 • ISSN No 2277 – 8179.*

[6] *Niroshinie Fernando , Seng W. Loke, Wenny Rahayu "Mobile cloud computing: A survey".*

[7] *Hoang T. Dinh, Chonho Lee, Dusit Niyato, and Ping Wang "A Survey of Mobile Cloud Computing: Architecture, Applications, and Approaches".*

[8] *Preeti Garg, Vineet Sharma "A Survey on Mobile Cloud Computing" November 2012.*

[9] *Bo Zhao, Zhi Xu, Caixia Chi, Sencun Zhu, Guohong Cao "Mirroring Smartphones For Good".*

[10] *Eric Chen1, Satoshi Ogata2, Keitaro Horikawa1 "Offloading Android Applications to the Cloud without Customizing Android" (23 March 2012 .*

[11] *Mayank Arora and Mala Kalra, "Autonomous Computation Offloading Application for Android Phones using Cloud ", In Proc Of SocProS 2012 Springer.*

[12] *Sumandeep Kaur*, Sugandha Sharma , Mayank Arora "Data Synchronization between Mobile and Cloud" Volume 4, Issue 6, June 2014 .*

[13] *Sudha S and Brindha K, Sai Vamsy Krishna S, Gokul K and Sanath Kumar M " Data Synchronization Using Cloud Storage" Volume 2, Issue 11, November 2012.*

[14] *B.G. Chun, P. Maniatis, " Augmented Smartphone Applications through Clone Cloud Execution," in Proceedings of 12th conference on Hot topics in Operating systems, 2009, pp. 8-8.*

[15] *Variza Negi1, Mala Kalra"Optimizing Battery Utilization and Reducing Time Consumption in Smartphones Exploitingthe Power of Cloud Computing".*

[16] *A. Rudenko, P. Reiher, G. J. Popek, and G. H. Kuenning: Saving portable computer battery power through remote process execution, vol. 2, no. 1, January 1998.*

[17] *Eduardo Cuervo, Aruna Balasubramanian, Dae-ki Cho, Alec Wolman, Stefan Saroiu, Ranveer Chandra, Paramvir Bahl: June 2010, MAUI-Making Smartphones Last Longer with Code Offload, pp. 49-50.*

[18] *Aki Saarinen, Matti Siekkinen, Yu Xiao, Jukka K. Nurminen, Matti Kemppainen "SmartDiet: Offloading Popular Apps to Save Energy".*

Author Profile

Er. SUMANDEEP KAUR received B,Tech. degree in computer science and technology from CT College of Engineering and Technology (PTU) and now pursuing M.Tech (CSE) from CGC Gharuan (Mohali). Research interest includes cloud computing.