**Review of Offloading Techniques over Cloud**

Hrishikesh Kore, Shweta Notiyal

Department of Electrical and Computer Engineering

Texas A&M University, Kingsville

(hrishikesh\_surendra.kore, shweta.notiyal)@students.tamuk.edu

**Abstract: Limited resources on mobile platforms compel us to balance available resources to maximize efficiency in mobile devices. Offloading is one such method done to reduce burden on devices by processing local computations remotely on a cloud-based services. Since the workload is distributed partially on the cloud, it saves energy and complex processing on mobile device. However, due to computation at remote locations, responsibility of data integrity and security lies with service providers who often fall short of doing so. To overcome this, various authors have proposed diverse techniques to better facilitate the secure migration of data when offloading between cloud and personal devices. This often entails an overhead of communication to extract and send data to compute remotely. But the benefits of offloading outweigh the communication costs to provide dynamic services to reduce stress of mobile applications on their host devices.**

**Keywords: SLA (Software Level Agreement), QoS (Quality of Service), MCC (Mobile Cloud Computing)**

**I. INTRODUCTION**

The use of mobile devices has spiked exponentially in the past few years. In recent statistics it is shown that nearly 90% of gadgets out there are smart phones which have become an integral part of human lifestyle. Faster computation has always been a trending topic among app developers who want to provide faster services as well as energy efficient ones. Eternally increasing computation speed with heavy processors is not the answer as it was in previous decades. The consumer demands the *best of all worlds* when it comes to mobile devices, their durability and ability to react instantaneously in day-to-day tasks. Limitations in these personal devices are battery capacity, efficiency, storage related constraints and processing power. Due to these reasons handheld devices cannot render huge tasks locally on the embedded system and need to

delegate computations to external resources, in this case: **Clouds.**

Clouds have the potential to provide high speed dynamic service to users with a simple internet connection. They have various services, depending on the user need and preference. SaaS (software as a service), PaaS (platform as a service), IaaS (infrastructure as a service) and the newly introduced XaaS (everything as a service) which hints towards an era in mobile computing where any functionality will be provided by clouds. In this paper, we have summarized various techniques of how to combine the idea of offloading in android devices with cloud computing.

The idea here is to partially upload computation from the mobile device for it to be processed remotely on the cloud. This reduces taxing the device resources and the illusion is seamless to the user as to where the computation is being carried out. An example of this is the Microsoft Office 365. It gives all functionality of *Office,* without installing any applications or widgets. One can simply access all services from the browser and get things done. Everyday more and more services like these are presenting themselves on clouds. This transparency is what we are trying to achieve in the long run which will enable us to virtually process even the heaviest of tasks on mobile devices.

Cloud is the umbrella term for further trending and useful technologies that have emerged through this decade viz. Fog, Cloudlet, Edge, etc. These extensions of the cloud push the boundaries and overcome challenges that are posed by the traditional cloud technology. Nowhere is the power or authority of the cloud undermined. It has its own place in high level batch processing and to date, remains as the backbone of newer technologies that have been tailored from the big cloth, i.e. Cloud.

The newly coined terms Fog and Edge represent virtual instances of the cloud services but at the edge of the network. The edge here refers to the proximity to the edge devices (user devices) and aims to provide faster and transparent execution capabilities to devices on the go. There are slight differences between edge and fog computing which are highlighted further in this paper.

The main aim here is to put ***offloading*** into the spotlight. There are innumerable scenarios where a user may want to offload data for processing, either implicitly or explicitly. Taking the same Office 365 example we can see that such documents are substantially large in terms of size or execution and hence, unlock the potential for offloading from a device. The way in which it is done can vary drastically, depending on available resource pool on devices and cloud capabilities.

Most of the offloading techniques proposed use an application which needed to be installed on board, which made the offloading decision for the other applications. The installed application is going to be heavy and will consume processing power of the device. This means that there is additional computational and communication overhead in each situation when an opportunity shows up in offloading. Other techniques proposed alternatives that determined the offloading criteria without such bloatware and were preferred when it came to performance and energy efficiency. Some of these are mentioned clearly in the next section.

**II. RELATED WORK**

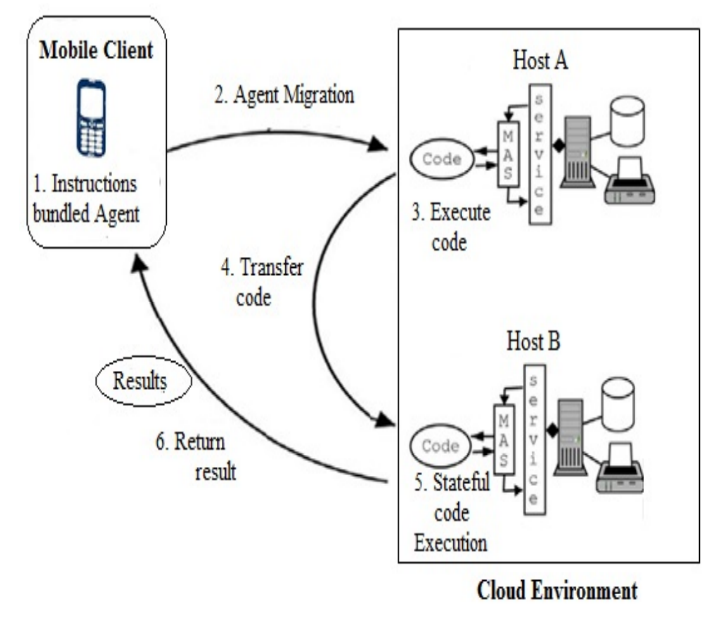
The terms cloud/fog/mist/edge have flooded the internet in recent years. The underlying driving force is predominantly computational offloading and various methods have been proposed over the years to efficiently offload data to-and-fro from applications to datacenters. All of them stand firm on the fact that resource limitation on mobile devices have been the seed for their research and proposed diverse techniques on those fronts.

Deqian KONG et. al proposed a dynamic offloading technique which revised the Android applications into two parts, one part executes locally and other parts (complex computational task) runs on the more resource-rich remote server which provides strong processing ability and less power consumption [1]. David I. Fadaraliki and S.Rajendran presented a mobile-based agents called JADE, it enhances the capabilities of mobile devices by sending the bundle of code and state to remote location server [2]. Harsh Bandhu Parnami et. al presented a research and related survey that a separate application needed to offload applications by making changes in the source code while developing an application [4]. Alessandro Zanni et. al demonstrated a demo which presents a tool, given a compiled APK locally. The methods enabled to be offloaded at runtime are chosen depending on computing time and energy usage, which are predicted in our framework. [7]. Romulo Reis de Oliveira et. al proposed a new technique that transparently includes code offloading capabilities to Android devices [8]. Quang-Huy Nguyen et. al introduced a hardware chip onto mobile devices to help calculate computational cost of applications and service [9]. Xiangyu Wu et. al. proposed a software-based approach that supports dynamic sensing and computational offloading on android devices; it pooled all sensor resources in the vicinity to extract accurate results in case other sensors fail [11]. Mayank Arora et. al. proposed an autonomous computation offloading framework for android (ACOF) using cloud; it empowered applications to run applications virtually on an image of the android phone on the cloud [12].

**III. OFFLOADING TECHNIQUES**

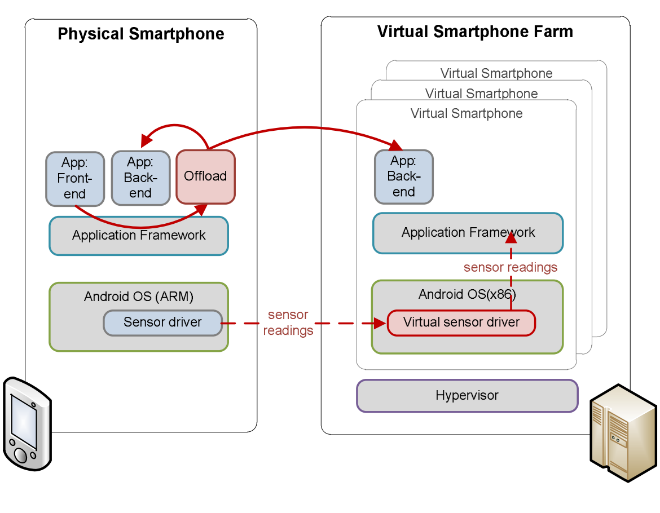
Some frameworks push the offloading framework to the extremes. [2] and [13] have showcased unique techniques that work specifically for the android platform.

1) The technology called JADE is primarily java-based and offers to offload application to run on resource rich computer far away from data source. A mobile agent is responsible to authenticate users. Then the data is wrapped in the agent (data, code, instructions) and sent over the network to the cloud environment for processing. The agents save the state of the data and are also responsible to transmit output data to the data source (device).



**Figure1: Proposed Architecture with JADE [2]**

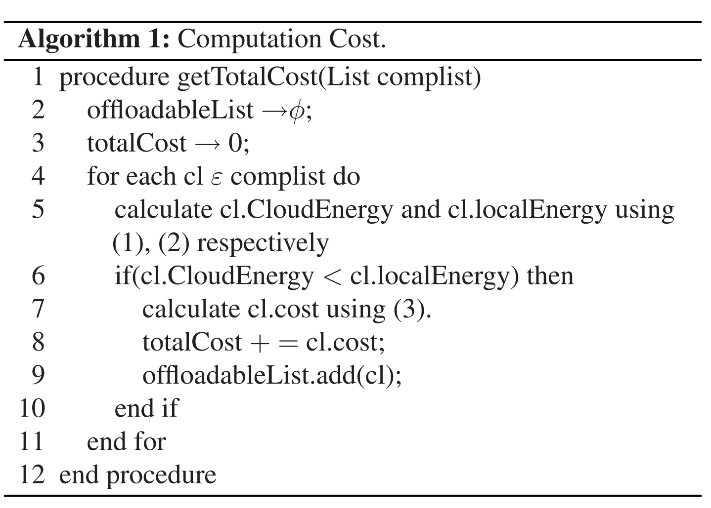
2) In reference [13] Eric Chen, et. al. proposes an offloading framework where they create a snapshot of the android device on the cloud, then test and compute the offloaded data on the virtual device. This allows the environment to learn which applications or part of applications are fruitful when offloaded and which are not so efficient and will cause excess resource usage than local, on-device computation.



**Figure2: Virtual Smartphone Architecture [13]**

In this paper, they demonstrated the design of a new offloading framework that enables an Android device to offload resource intensive work to a server in the cloud datacenter. The design decisions were determined by the intention to deploy the architecture in the application layer without modifying the underlying stock Android platform and the application source code.

3) The most interesting proposed methodology is showcased in [15] where Jitender Kumar et. al. proposed “*Demand-Based Computation Offloading Framework for Mobile Devices”.*  Jitender Kumar et. al used demand-based framework by using extended CloudSim simulator for resolving a major issue of application partitioning according to user’s demand. As previous technologies have focused on the issue of saving energy in the entire offloading process and no other, the authors here address the issue that the amount of computation at the user end should also be taken into consideration along with providing a framework where application partitioning is done dynamically on user-demand while maintaining QoS and meeting SLAs.



**Algorithm 1: Demand-Based Offloading Algorithm [15]**

This algorithm allows any number of requests to be initiated by the mobile device and they are queued to be computed on cloud if the energy required for computation is lesser when offloaded, i.e. energy to compute on cloud is less than that required for local computation. Various simulations run on CloudSim proved that this architectural setup reduced the cost of cloud provisioning to mobile devices by almost 45% and minimized the SLA violations by 1.34% for maintaining the Quality of Service.

**IV. FUTURE SCOPE**

The chart below highlights the fact that many challenges such as security have not been handled in the whole offloading dynamic and many more remain undiscovered as of now. More research is being done day by day to understand such issues and tackle them as and when they come up. We can only say that it is exciting to see the community at work to achieve better computational possibilities in different environments, especially when it comes to offloading on Android and other mobile devices.

**V. CONCLUSION**

We reviewed a variety of papers on offloading on mobile devices which is a trend in this era. The technological advancements have been tremendous and continue to tackle one issue at a time to achieve a better offloading framework, each time. It is obvious to say that there cannot be one *perfect* method to achieve this as it depends on the user and application demands. However, new paradigms in cloud and

**Figure 3: Comparison of Areas Focused in Offloading**

mobile computing has enabled an easy transition for the offloading world to provide seamless resources for real-time, on-demand, low latency computation on mobile devices. These technologies work in unison to create a pervasive computing environment that ubiquitously caters to the need of the users, anytime, anywhere.

**REFERENCES**

# [1] Deqian Kong, [Tao Qi](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Tao%20Qi.QT.&newsearch=true), [Tan Yang](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Tan%20Yang.QT.&newsearch=true), [Yidong Cui](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Yidong%20Cui.QT.&newsearch=true)., “A new dynamic computation offloading framework for Android”, IEEE, [10.1109/ICBNMT.2013.6823930](https://doi.org/10.1109/ICBNMT.2013.6823930)

[2] [David I. Fadaraliki](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.David%20I.%20Fadaraliki.QT.&newsearch=true); S. Rajendran, “Process offloading from android device to cloud using JADE”, 2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015], Year: 2015

# [3] [Abdullah Yousafzaia](https://www.sciencedirect.com/science/article/pii/S0164121216301364#!) [Abdullah Gania](https://www.sciencedirect.com/science/article/pii/S0164121216301364#!) [Rafidah Md Noora](https://www.sciencedirect.com/science/article/pii/S0164121216301364#!) Anjum NaveedaRaja Wasim Ahmada[Victor Changb](https://www.sciencedirect.com/science/article/pii/S0164121216301364#!), “Computational offloading mechanism for android native and runtime based mobile applications”

[4] Harsh Bandhu Parnami , Deepika Khokhar, Mayank “Arora Framework for offloading necessary android Applications using Cloud”

[5] Roelof Kemp, Nicholas Palmer, Thilo Kielmann and Henri Bal, “A Computation Offloading Framework for Smartphones”,https://link.springer.com/chapter/10.1007/978-3-642-29336-8\_4

[6] Android Applications Offloading Using Cloud Computing - Harsh Bandhu Parnami, Mayank Arora, DeepikaKhokhar,www.pdfs.semanticscholar.org/edb4/fb88c887ed5e7409f0a0a63177795a7304d1.pdf

## [7] [Se-young Yu](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Se-young%20Yu.QT.&newsearch=true);  Dr.[Stefano Secci](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Stefano%20Secci.QT.&newsearch=true); [Rami Langar](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Rami%20Langar.QT.&newsearch=true); [Paolo Bellavista](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Paolo%20Bellavista.QT.&newsearch=true); [DanielF.Macedo](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Daniel%20F.%20Macedo.QT.&newsearch=true),”Automated offloading of android applications for computation/energy optimizations”,2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)

## [8] A transparent code offloading technique for Android devices - 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)

## [9] [Quang-Huy Nguyen](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Quang-Huy%20Nguyen.QT.&newsearch=true); [Johannes Blobel](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Johannes%20Blobel.QT.&newsearch=true); [Falko Dressler](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Falko%20Dressler.QT.&newsearch=true), Energy Consumption Measurements as basis for Computational Offloading for Android Smartphones, 2016 IEEE Intl Conference on Computational Science and Engineering (CSE), Pages: 24 – 31

## [10] [Gabriel Orsini](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Gabriel%20Orsini.QT.&newsearch=true); [Dirk Bade](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Dirk%20Bade.QT.&newsearch=true); [Winfried Lamersdorf](https://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Winfried%20Lamersdorf.QT.&newsearch=true), “Mobile Edge: Designing Elastic Android Applications for Computation Offloading”, 2015 8th IFIP Wireless and Mobile  Networking Conference (WMNC).

## [11] CoseDroid: Effective Computation- and Sensing-Offloading for AndroidApps.

## [12][MalaKalra](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Mala%20Kalra.QT.&newsearch=true); [SarabjeetSingh](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Sarabjeet%20Singh.QT.&newsearch=true),”Computation offloading or Android using cloud Mayank Arora; , 2013 2nd International Conference on Information Management in the Knowledge Economy

## [13] Offloading Android applications to the cloud without customizing Android

[Eric Chen](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Eric%20Chen.QT.&newsearch=true); [Satoshi Ogata](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Satoshi%20Ogata.QT.&newsearch=true); [Keitaro Horikawa](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Keitaro%20Horikawa.QT.&newsearch=true)

[2012 IEEE International Conference on Pervasive Computing and Communications Workshops](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/xpl/mostRecentIssue.jsp?punumber=6192378), Year: 2012

## [14] A transparent code offloading technique for Android devices, [Rômulo Reis de Oliveira](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.R%C3%B4mulo%20Reis%20de%20Oliveira.QT.&newsearch=true); [Nícolas Meira da Silva Schirmer](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.N%C3%ADcolas%20Meira%20da%20Silva%20Schirmer.QT.&newsearch=true); [Mateus Machry](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Mateus%20Machry.QT.&newsearch=true); [Tiago Coelho Ferreto](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Tiago%20Coelho%20Ferreto.QT.&newsearch=true)

2017 13th International Wireless Conference (IWCMC)

## [15]Demand-Based computation Offloading Framework for Mobile Devices [Jitender Kumar](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Jitender%20Kumar.QT.&newsearch=true); [Amita Malik](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Amita%20Malik.QT.&newsearch=true); [Sanjay K. Dhurandher](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Sanjay%20K.%20Dhurandher.QT.&newsearch=true); [Petros Nicopolitidis](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Petros%20Nicopolitidis.QT.&newsearch=true) IEEE Systems Journal

Year: 2017, ( Early Access )

## [16] Partitioning Android mobile application for improving execution time, [Kailas K. Devadkar](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Kailas%20K.%20Devadkar.QT.&newsearch=true); [Dhananjay R. Kalbande](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Dhananjay%20R.%20Kalbande.QT.&newsearch=true), [2015 IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/xpl/mostRecentIssue.jsp?punumber=7428302)

## [17] [Understanding the Challenges in Mobile Computation Offloading to Cloud through Experimentation](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/document/7283056/) [Padmaja Joshi](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Padmaja%20Joshi.QT.&newsearch=true); [Ashwin Nivangune](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Ashwin%20Nivangune.QT.&newsearch=true); [Ranjan Kumar](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Ranjan%20Kumar.QT.&newsearch=true); [Sathish Kumar](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Sathish%20Kumar.QT.&newsearch=true); [Rakesh Ramesh](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Rakesh%20Ramesh.QT.&newsearch=true); [Sushant Pani](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Sushant%20Pani.QT.&newsearch=true); [Arif Chesum](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/search/searchresult.jsp?searchWithin=%22Authors%22:.QT.Arif%20Chesum.QT.&newsearch=true), [2015 2nd ACM International Conference on Mobile Software Engineering and Systems](https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/xpl/mostRecentIssue.jsp?punumber=7277122), Pages: 158 – 159

[18] Offloading Android Applications to the Cloud without Customizing Android - <https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/stamp/stamp.jsp?tp=&arnumber=6197619>

[19]   ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading - <https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/stamp/stamp.jsp?tp=&arnumber=6195845>

[20] Pervasive Computing: Vision and Challenges - <https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/stamp/stamp.jsp?tp=&arnumber=943998>

[21] Mobile Cloud Computing Based Energy Efficient Offloading Strategies for Femtocell Network - <https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/stamp/stamp.jsp?tp=&arnumber=6785515>

[22] A Profile-Driven Dynamic Application Offloading Scheme for Android Systems - <https://0-ieeexplore-ieee-org.oasis.lib.tamuk.edu/stamp/stamp.jsp?tp=&arnumber=6379903>

[23]  Computing at the Mobile Edge: Designing Elastic Android Applications for Computation Offloading.

[24] Tendai Munyaradzi Marengereke, K. Sornalakshmi, Cloud Based Security Solution for Android Smartphones, 2015 International Conference on Circuit, Power and Computing Technologies [ICCPCT]