ISSN No:-2456-2165

A Survey on Efficient Mobile Cloud Computing Through Computational Offloading

Kapse Preeti
Department of Computer Engineering
Shri Chatrapati Shivaji Maharaj College of Engineering
Nepti, Ahmednagar, India

Bangar P. H.
Prof., Department of Computer Engineering
Shri Chatrapati Shivaji Maharaj College of Engineering
Nepti, Ahmednagar, India

Abstract:- Mobile cloud computing (MCC) represented a new era in computing, in which cloud users are drawn to a variety of services over the Internet. MCC has a high-quality, adaptable, and cheap delivery platform for providing Internet-based services to mobile cloud users. The rapid growth of data produced by various smart devices, such as smart phones, IoT/IIoT networks, and vehicular networks running various specific applications such as Augmented Reality (AR), Virtual Reality (VR), and positioning systems, necessitates an increase in processing and storage resources. Offloading is a promising strategy for dealing with such devices' intrinsic limitations by transferring resource-intensive code, or at least a portion of it, to neighbouring resource-rich servers. This paper explains how to use offloading techniques to make mobile cloud computing more efficient, as well as the framework for doing so.

Keywords:- *Computation Offloading, Mobile Cloud Computing, Mobile Edge Computing, Offloading Framework, Algorithm, Mobile Applications.*

I. INTRODUCTION

Mobile gadgets, such as smartphones and PDAs, have advanced dramatically in recent years. With the advancement of mobile applications, users are no longer restricted to using their phones for calling and sending SMS.Smartphone users can detect the environment, develop social interactions, and use easy apps (e.g., health care, e-commerce, mobile learning) thanks to the multi-dimensional and beneficial application markets (Apples iTunes, Google Play, Nokia's Ovisuite).Due to resource constraints such as finite battery energy, poor CPU speed, insufficient storage space, poor network bandwidth, and poor sensing capacity, it is still difficult to construct particularly sophisticated apps on mobile devices . In addition, to reduce device fragmentation , almost all smartphones use cross-platform runtime environments to develop and run applications, such as Java ME,.NET CF, and Android. As a result, concerns such as mobility management, quality of service insurance, energy management, and security are brought to the forefront.

In various reasearch papers and potential reffernces I used for this survey, various solutions for compute offloading in mobile contexts rely on cloud computing infrastructures. Cloud computing fulfils the vision of computing as a utility (similar to water, electricity, gas, and telecommunications) by providing a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, and applications) that can be rapidly provisioned and released. However, in a mobile context, standard cloud infrastructures might cause severe network delays that detract from the user experience and overshadow the solution's potential benefits.

For these reasons, mobile devices have incorporated the concept of workload offloading to the cloud. Many difficulties, including as security, quality of service, and mobile application development, are overcome when a work is offloaded to the cloud. Complex computations are required by applications such as real-time computing, image processing, voice processing, online games, video streaming, language, and wearable computing. In terms of deploying the application for mobile devices, these complicated apps pose a challenge for mobile device application developers. Because of the increased demand for smaller sized devices, the problem of battery power consumption and memory space is becoming more of an issue.

The general concept is to computational offloading to the cloud. Because of the limited processing speed and setup of mobile devices, processing heavy jobs are difficult to run. Furthermore, when running time-consuming applications, the battery limit is a major worry. The concept is to move code areas to the cloud. Because of its powerful and huge CPUs, the cloud can run tasks more efficiently and swiftly than a mobile device. In recent years, offloading techniques have made it easier to create computing-intensive apps. However, issues such as heterogeneity of the system, physically distributed mobile devices and cloud, the inconsistency of data transmission rate have made the offloading decision difficult in a different environment.However, in a new setting, challenges such as system heterogeneity, physically distant mobile devices and cloud, and data transmission rate inconsistency have made the offloading choices problematic.

Thr remaining part of my survey paper is structured as following: section (i) Literature Survey of the related research, survey and study papers and articles. Section (ii) System Design: It explains the basic architecture of cloud computing, mobile cloud computing, architecture and farmeworks og computational offloading. Section (iii) Implementation of the proposed architecture and efficient technique of the computational offloading. Section (iv) Results and outcomes of the techniques anf frameworks applied. Section (v)Conclusion and analysis of the survey.

II. LITERATURE SURVEY

As I came across a variety of literature such as research papers, published work by authors and researchers, surveys, and case studies on the subject of Efficient Mobile Cloud Computing through various approaches focusing on Computational offloading, its frameworks, mechanisms, models, architectures, and challenges illation can be drawn as it is a vast field to dive into.

While researching the subject, I came upon a news release from the Future Generation Computer System. The study paper was titled "quick-response framework for computational offloading" and was written by China's Southern Technical University. They suggested a compute offloading architecture based on agents with low request response latency.They considered the job completion time limitation and connection bandwidth limit in the framework, with the goal of maximising energy savings for all devices , making the offloading technique more practical. They also converted the problem into a standard 0–1 Knapsack problem. They presented a Dynamic programming after filtering (DPAF) algorithm to handle the problem, which can cut energy usage as much as feasible while meeting completion time and link bandwidth limits. The filtering method we use, in particular, can significantly minimise the number of offloading requests, which helps to simplify the DPAF algorithm.In terms of energy savings and request response delay, their offloading framework outperforms other offloading schemes or all local execution systems, according to simulation data. In addition, their DPAF technique saves more energy than competing offloading techniques.

In the based paper i choose for this survey “Efficient Mobile Cloud Computing through Computational Offloading” (Biswas M, Whaiduzzaman MD (2019) Efficient Mobile Cloud Computing through Computation Offloading . Int J Adv Technol 10: 225. doi:10.4172/0976-4860.1000225)authors have given various approaches to attent the efficiency and level of ease in mobile cloud computong using offloading framework, alogorithms and mathematical models.Mobile Message Passing Interface (MMPI), Virtual machine (VM) migration, MobiCloud, Grid model, Cuckoo, MAUI, Clone cloud these are some Android frameworks explained by the authors.

There are two components to their future work. First, they want to think about using the cloudlet system to save energy for several users. Second, they'll talk about a more difficult task offloading connection.

III. SYSTEM DESIGN

MOBILE CLOUD COMPUTING

MCC is a new revolution that moves data processing and storage from the mobile device to the cloud, reducing the mobile device's processing power and energy consumption. The generated data is transmitted back to the device after the processing is completed on the cloud server. With recent advancements in data transmission rates, centralised cloud server access has become simple. Cloud servers are vast and readily available, as they offer all cloud computing features. As a result, MCC is a hybrid of mobile computing and cloud computing. This technology combination has alleviated the limitations of mobile computing. With the advent of MCC, customers may now access a variety of services using powerful applications on their mobile device.

Mobile devices can connect to the cloud server in two ways in a mobile cloud networking architecture (e.g., through mobile network or through access point network). The mobile devices are connected to the mobile network providers via base stations in the case of a mobile network. The connection speed, link capacity, and functional interfaces between the network and mobile users are all controlled by the base station (e.g., transceiver station, access point, satellite). The central processor, which is connected to the server, receives the user's requests.On the basis of stored information in the database, this server maintains the service of a mobile network, including authorization, authentication, and account validation. The queries are then transferred to the clouds, where they are processed, and the resulting data is returned over the mobile network. Cloud servers develop utility computing, virtualization, and service-oriented architecture.

In the case of an access point, mobile devices connect to the access point via a closer Wi-Fi network, which is then connected to the Internet Service Provider (ISP) to give internet access. Users can use cloud services without having to use a telecommunications network, which may charge for data transmission. Furthermore, telecom network connectivity (e.g., 3G/HSDPA or 4G) consumes more energy than access point connectivity. As a result, when Wi-Fi connectivity is available, mobile devices prefer to use it to access cloud services.

The cloud computing architecture might vary depending on the situation. For instance, a web-delivered business-oriented architecture was developed for constructing market-oriented clouds. In, a four-layer design is given, and cloud computing and grid computing are compared. The layered architecture of CC was the subject of this work. In terms of usability and efficiency, this design best describes CC.

ISSN No:-2456-2165

COMPUTATIONAL OFFLOADING

The general concept is to offload the data to the cloud. Because of the limited processing speed and setup of mobile devices, processing heavy jobs are difficult to run. Furthermore, when running time-consuming programmes, the battery limit is a major worry. The goal is to get the code out of the way. Because of its powerful and huge CPUs, the cloud can run tasks more efficiently and swiftly than a mobile device. In recent years, offloading techniques have made it easier to create computing-intensive apps. However, in a new setting, challenges such as system heterogeneity, physically distant mobile devices and cloud, and data transmission rate inconsistency have made the offloading choices problematic.

Offloading decision-making: A lot of factors can influence the decision to offload a calculation. Unloading little pieces of code might sometimes be more energy and time consuming. Then computing that area locally is a smart idea. The user may not want to offload because of the data charge.

As a result, decision-making and the development of a convenient offloading mechanism are essential for successful mobile cloud computing.

A mobile cloud application follows the procedures outlined in the Offloading work flow diagram. The application first determines whether the execution will be statically or dynamically. The portion of the code that will be offloaded in a static process is predefined. As a result of the code region separation, some jobs are processed locally while others are offloaded. There is no change in the offloading choice with a change in connection speed or any process effect in static offloading. Dynamic offloading, on the other hand, makes offloading decisions based on current environmental conditions. The favourite component is offloaded to the cloud if offloading is enabled and the resource is available in the cloud. Aside from that, the code is performed on mobile devices. Offloading can be done in a variety of ways. To establish an efficient offloading method, several sorts of research have been conducted. Various offloading strategies correspond to various network architectures, device configurations, and protocols. Different offloading strategies can be categorised into three categories: Client-server communication, virtualization, and mobile agents are all examples of client-server communication.

Various aspects, such as performance enhancement and energy consumption minimization, must be considered while making the compute offloading decision. Before computation offloading, a number of questions must be answered, such as:

Is it possible to offload this task? The task scheduler must assess whether the task can be offloaded, and if so, whether it should be offloaded in part or whole.

1) When is it appropriate to delegate responsibility for a task? Under various limitations, the task scheduler must decide the offloading time slot.

It's critical to select which parts of a task should be offloaded to edge devices and which should be offloaded to the cloud data centre when working with edge-cloud collaboration. Before computing collaboration between edge devices and cloud data centres, the initial step is computation partitioning. Applications stored in cloud data centres can be transferred to globally distributed edge servers and edge nodes as the computational capacity of edge devices grows. Partitioning an application is the process of breaking it down into smaller pieces.

2) What is the best place to offload? According to the distribution of available resources, the query is interpreted as "which location is ideal for offloaded job execution?"

Computation offloading allows computation-intensive applications and services to take advantage of the compute, storage, networking, and energy capabilities of edge devices while reducing latency. However, because network conditions change dynamically during programme execution, the choice on burden offloading must be made at the appropriate time. To put it another way, the task scheduler must carefully time the offload opportunity while taking into account all conditions and system state.

3) Which policy for offload will be used? That example, is the primary goal of workload offloading to maximise a particular performance metric or to optimise and tradeoff many goals?

Smart mobile devices near the network's edge typically have limited resources, such as computational power, storage space, and battery capacity, making it challenging to accommodate mobile users' expanding demands. Resources must be allocated and scheduled according to user requirements and service level agreements in order to provide higher quality services (SLAs). As a result, time-sensitive applications must be prioritised, and computation-intensive programmes must be provided with sufficient computing resources.

Computational offloading Framework

To attain the most efficient form of MCC through Computational Offloading differnt reliable frameworks come in sight. There are several frameworks like Mobile Message Passing Interface (MMPI),CUCKOO.

Breif of Offloading frameworks:

Using a Java stub/proxy approach, the ‘Cuckoo' framework proposes a solution for offloading mobile device apps to the cloud. Cuckoo can be deployed on any resource that supports the Java Virtual Machine, such as Amazon EC2 10 or a private micro cloud made up of laptops and local clusters. Other mobile phones, on the other hand, have not been suggested as possible resource providers. Cuckoo's offloading goals for Android are to improve performance and decrease battery drain. Cuckoo's communication component is built on the Ibis High Performance Programming System.To use Cuckoo, the programmes must be rewritten in such a way that they can support both remote and local execution. Application developers are given access to a programming model that serves as the system's interface for this purpose. The programming approach is based on Android's existing "activity/service" concept, which distinguishes between services (computing-intensive techniques that can be offloaded) and activities (interactive methods of the application). At the activity, a proxy object is constructed that is linked to the actual implementation.

The Mobile Message Passing Interface (MMPI) framework is a Bluetooth-based mobile variant of the conventional MPI, in which mobile devices act as resource providers. MMPI has a completely interconnected mesh structure instead of the conventional star network structure of regular piconets, allowing each node to communicate with the others. Tasks like device discovery and connections are handled by the framework's libraries, which eliminates the need to write any Bluetooth-specific code explicitly.The framework is written in Java, and Bluetooth operations are handled by the Blue Cove third-party library. The slave devices get job parameters from the master mobile device, which they then execute. They used fractal generation as a test application on several devices (phones, computers, and PDAs) across platforms and documented the time cost. They do not, however, provide the cost of employing only one machine (in the traditional manner), therefore it is unclear what the speedup was.

A grid model is one in which a mobile device connects to a server as a thin client using a traditional thin client protocol like VNC (Virtual Network Computing) or a streaming protocol. User input is transferred to the server through the wireless network, and the server processes it before sending back the necessary graphical output, which is then displayed on the mobile device. This study focuses on server selection techniques that are required when the location of a mobile device changes.

Migration of virtual machines (VMs). VM migration is the process of moving a virtual machine's memory image from one server to another without interrupting its operation. In a live migration like this, the VM's memory pages are pre-copied without disrupting the OS or any of its programmes, giving the impression of a seamless transfer. Because the VM border insulates the surrogate device, this strategy assures that no code changes are required when applications are offloaded. It also guarantees a relatively secure execution.However, VM migration is time expensive, and the workload on mobile devices may be too much.

Satyanarayan proposes ‘cloudlets' as an alternative to connecting to a remote cloud. A cloudlet is a small data centre that is located in certain locations and connected to a bigger cloud server over the Internet. They describe a cloudlet as "internally resembling a cluster of multi-core machines with gigabit internal connection and a high-bandwidth wireless LAN."

Clone Cloud also employs virtual migration to offload a portion of the programme to the cloud through Wi-Fi or 3G. The programme does not need to be changed because a clone of the actual device is used for migration. Additionally, developers do not annotate methods like MAUI. Clone Cloud analyses the cost of offloading and executing remotely in comparison to local execution using a "cost model."

The MobiCloud model proposes a secure implementation of cloud computing technologies for MANETs (Mobile Ad hoc NETworks). MANETs can be transferred to a service-oriented framework using MobiCloud. Every mobile device in MobiCloud is designed to be a service provider or broker. These service nodes are virtualized components that are replicated in the cloud. Because the service nodes can be a clone, a partial clone, or merely an image of the real device, the Extended Semi-Shadow Images (ESSIs) aren't exactly the same as virtual machine migration.

VI. RESULT

With the evolvement of cloud computing, there has been opened many opportunities to build rich applications for mobile devices. Resource and energy problems have been reduced by cloud technology. Now the vendors are able to build new applications for smartphones which were almost impossible previously. Therefore, the global mobile application market is greatly influenced by mobile cloud computing. And using new efficient frameworks and algorithms MCC have changed it’s challenges into applications like augmented reality, mobile healthcare, mobile e-commerce, mobile gaming, web application.

Several in detail applications of MCC:

Mobile healthcare:

The goal of mobile healthcare is to provide better service to patients, clinicians, and hospitals. It will make medical care more affordable, simple, and error-free. Furthermore, it gives hospitals the option of using independent local server applications. In the universal healthcare app store, there are five mobile healthcare apps.

Health monitoring services allow users to connect with one other while also monitoring and recording the patient's condition. Emergency services can respond quickly to a medical emergency (such as a vehicle accident) by dispatching a medical team to the exact site. Pulse rate, blood pressure, sugar, and alcohol levels can all be detected by health-aware mobile devices, which then submit the data to a monitoring service. Users and clinicians will be able to access current and historical health information at any time and from any location thanks to the universal database. Universal lifestyle management can assist in the automatic payment of healthcare costs and other relevant charges.

Mobile commerce:

A business concept for commercial activity using a mobile phone is known as mobile commerce. The internet is now used for the majority of financial transactions. Because of its mobility and accessibility, mobile commerce can improve finance, advertising, and shopping. Using the capabilities of cloud computing, various secure applications (e.g., mobile payment, transaction, and ticketing) can be built for mobile devices. For mobile commerce, several difficulties such as network delay and security may occur. Yang proposes a mobile commerce architecture that combines the benefits of the 3G network with CC.

Online Learning:

Mobile devices can be a helpful tool for learning and staying in contact with enormous amounts of data. Traditional mobile learning required expensive equipment, long communication delays, and limited storage space. CC can widen the concept and allow for the development of rich mobile learning applications with high computing power, vast data storage, and low cost. Zhao uses a mixture of mobile learning and CC to improve student-teacher communication. Students communicate with teachers via a Google App Engine-based web application.

VII. CONCLUSION

More study may be done to establish a convenient strategy now that mobile devices are more capable because to MCC. The mobile cloud has the potential to change the technology trend and make a significant difference in our daily lives by solving a variety of real-world challenges. We introduced mobile cloud computing in this study. To make the best use of MCC, they presented offloading decision-making and offloading methodologies. Later, we discussed application models that could be used in future mobile apps and could alter the appearance of current apps.

REFERENCES

[1]. Efficient Mobile cloud computing through Computational offloading, Published in 2019 IEEE Cloud Computing conference, Author:Milon Biswa, Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka, Bangladesh, Md Whaiduzzaman, Institute of Information Technology, Jahangirnagar University, Savar, Dhaka, Bangladesh.
[2]. https://www.marketsandmarkets.com/Market-Report mobile-applications-228.html https://www.businesswire.com/news/home/20110609005403/en/
[3]. Worldwide-Smartphone-Market-Expected-Grow-55-2011.
[4]. Conti M, Chong S, Fdida S, Jia W, Karl H, et al. (2011) Research challenges towards the Future Internet. Comput commun 34: 2115-2134.
[5]. Gavalas D, Economou D (2010) Development platforms for mobile
[6]. applications: status and trends. IEEE Software 28: 77-86.
[7]. Satyanarayanan M, Bahl P, Caceres R, Davies N (2009) The case for VM- based cloudlets in mobile computing. IEEE Pervasive Comput 8: 14-23. Varshney U (2007) Pervasive healthcare and wireless health monitoring. J Mobile Net App 12: 113-127.
[8]. Yang X, Pan T, Shen J (2010) On 3G mobile e-commerce platform based on cloud computing. 3rd IEEE International Conference on Ubi-Media Computing.
[9]. Zhao W, Sun Y, Dai L (2010) Improving computer basis teaching through mobile communication and cloud computing technology. Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering.
[10]. Li Z, Wang C, Xu R (2001) Computation offloading to save energy on handheld devices: a partition scheme. Proceedings of the 2001 international conference on Compilers, architecture, and synthesis for embedded systems.