M. R. Sudha, International Journal of Computer Science and Mobile Applications,
Vol.6 Issue. 2, February- 2018, pg. 8-20 ISSN: 2321-8363
Impact Factor: 5.515

An Optimal Partitioning Algorithm of Mobile Device Interface to the Cloud Application Systems

M. R. Sudha*
*Assistant Professor, Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology, Kattankulathur-603203, Chennai, Tamil Nadu, India.
Email: sudha.mr@ktr.srmuniv.ac.in

Abstract: ***The mobile cloud computing technology has developed as a new information technology infrastructure for the fast-growing IT industry. In cloud computing, information is eternally saved in big scale data centers on the Internet using the whole world and for the moment approached and cached on clients including desktops and portable PCs, sensors, etc. With the "cloud" as a symbol for the Internet, cloud computing assures to distribute extremely amenable data, software, and hardware requirements as a service to external clients with Internet accesses. The cloud computing is the fundamental technology to achieve proficient in the operation of IT solutions. The data stored in cloud contacts and the execution program is moved to cloud domain. The mobile user is set free from resource forced issue of existing mobile devices. Moreover, to advantage the cloud services, the communications between mobile devices and clouds are supposed through the wireless medium. So, new programmes of security and privacy challenges introduced. The purpose of this survey is to present the safety and seclusion challenges in this field which have developed much attention among the academic and research community. Strengthen by mobile cloud computing, previously unrealizable mobile applications are resulting from their way into mobile devices. These emerging applications range from portable storage and authentication and fast internet to synergistic affluent media services. However, the running and extensive operation of these requests confront various technical challenges across the different layers of wireless networks. The main difficulties of mobile cloud computing include as the improvement of next-generation mobile cloud computing services and appliances. The energetic distribution algorithms for smart devices linked to mobile cloud services, the plan of admittance and contact schedules for cloud computing in a mobile environment and the necessary augments in mobile cloud computing to make possible mobile rich media. In this paper, we presented a comprehensive survey of mobile cloud computing research, as the importance the specific interests in mobile cloud computing besides with various difficulties to be measured and their solutions.***

Keywords: Mobile Cloud Computing, Mobile Devices, Applications, Algorithm

1. INTRODUCTION

The fabulous detonation of mobile devices is future from over. Smartphones and tablets are at the center of a new present upheaval are enchanting mobile computing to a whole new level. These devices grow to be still smarter with the beginning of innovative and compelling services that can take improvement of the computational possessions on both the device itself and on the cloud. Apple Siri, Google voice search, sensing and communication aptitudes are approaching the new class of devices near the awareness, one day, of a determined grand vision: all-in-one smart, powerful, and multitalented devices the ultimate combination of phones, tablets, and personal computers. It is not also hard to envisage a world where we will be able to take a single

device to answer phone calls, use it the primary personal computer at work by fixing it up to external monitors, keyboards, and set up desired games as much as we do at present on tablets and smartphones. The future evaluation of the rising number is still optimistic anticipates mobile devices with phones best the number of population on earth very soon [1]. Mobile cloud computing set-up presented in Figure 1.

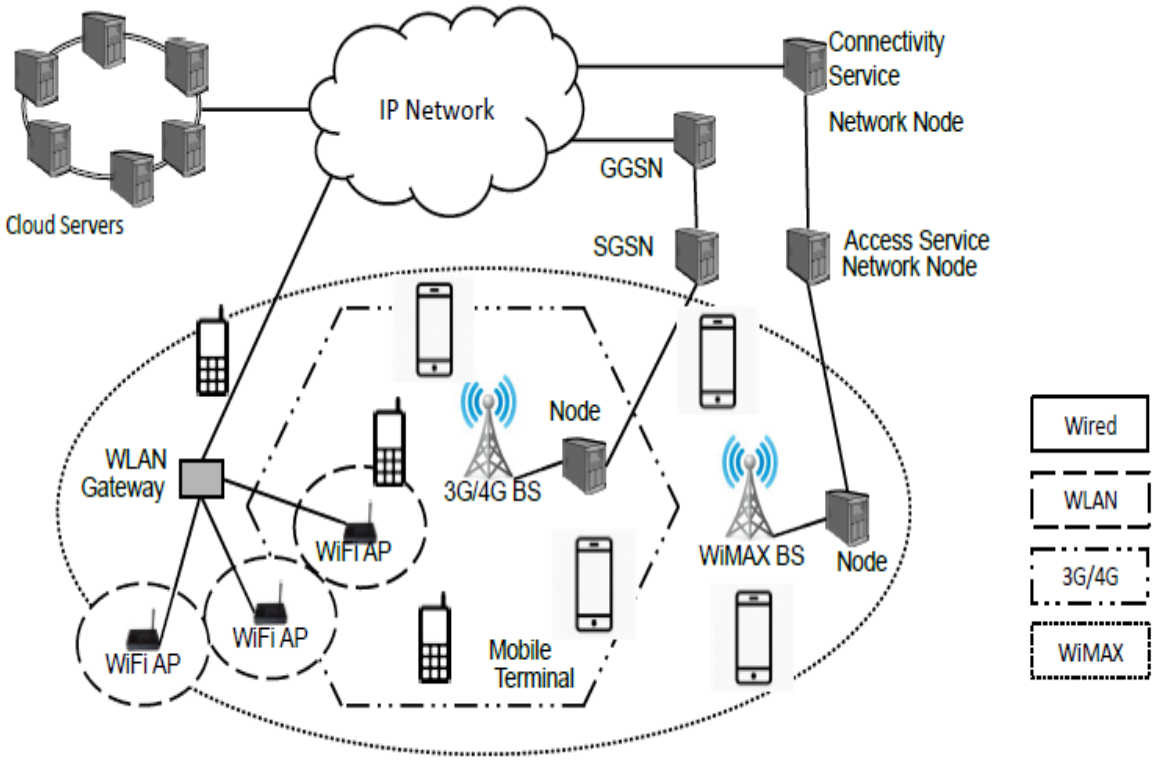

Fig. 1. Mobile cloud computing uses everywhere mobile phones connect to cloud servers via various network interfaces such as WiFi, cellular network, and WiMax

We surveyed that significant efforts can be found in results to deal with the challenges of estimated power and battery lifetime by offloading computing tasks to the cloud. In the Many researchers have proposed several problems such as Client server offloading [2, 3], peer-to-peer offloading [4], code offloading [5], Virtual machine migration [6], Cloudlets [7], Clone Cloud [8], Mobile Cloud [9], and more recently application server based parallel offloading [10]. A schematic representation of the development of these devices' characteristics is shown in Figure 2. Despite the fact that computation and storage most likely continue to augment according to Moore's law, it is not clear if the cellular communication channel and the battery capacity present similar trends. Battery still ruins restricted access, not being able to deal with larger screens, computation, and data exchange needs of many applications, also whilst computation and memory likely amplify significantly, network bandwidth and battery capacity not develop at the same rate upcoming with the device itself or from the marketplaces.

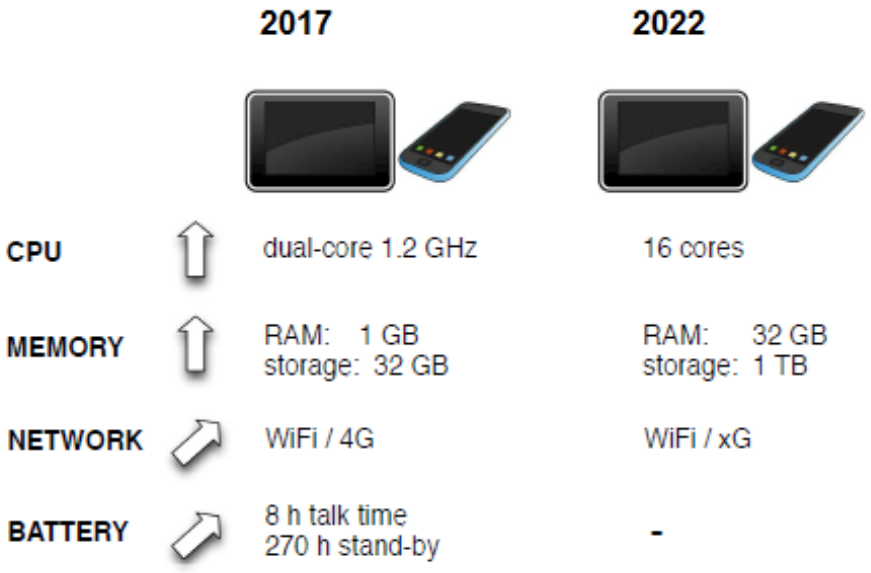

Fig. 2. Mobile devices are today and in five years from now

The cellular wireless bandwidth strength not reveals extensive upgrades in the short term and often a new wireless technology rapidly practices requirement of capacity particular the constant traffic development due to the unstoppable demand for new data applications. It is the case of applications that, at present heavily depend on the cloud to carry out their tasks speech and image processing. The paper structured into the following sections. Section 2 presents the fundamental concepts of MCC, cloud-based mobile applications, implementations in MCC and execution optimization. Section 3 discusses the utilization of partitioning algorithms. Section 4 describes the security confront in MCC. The open challenges explained Section 5. Finally, we conclude the future research work presented in section 6.

2. FUNDAMENTAL CONCEPTS OF MCC

The promising technology of integrating Mobiles with cloud computing builds a new programming concept called Mobile Cloud Computing (MCC). MCC guarantee to overcome mobile restrictions as the processing and storage for intensive works moved to the cloud to obtain position there and the final results arrival back to the mobile device. Mobile cloud computing defined as follows "Mobile Cloud Computing (MCC) measured as a kind of cloud computing with support computing for mobility such as location awareness, computing capability, and data backup" [11]. As in figure 3, mobile devices in MCC mainly use the 3G/LTE to join to the cloud and seldom use the Wi-Fi since it is not for all time accessible, table 1 compares between these two connection technologies.

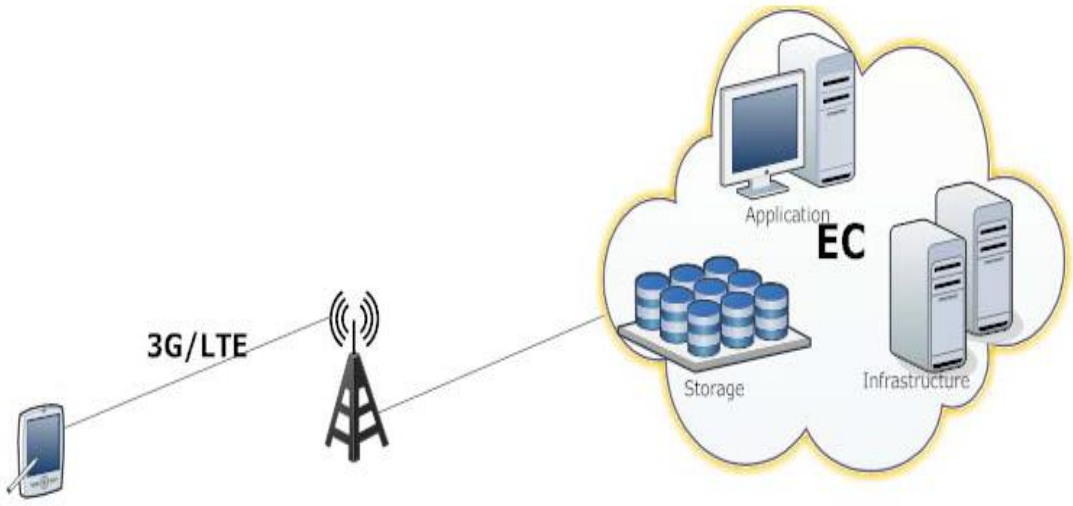

Fig. 3. Traditional MCC Architecture

From table 1 explained that using Wi-Fi connection is desirable over the 3G/LTE connections. Cloudlet [7] is a real and high potentials computer or cluster of computers, connected to the internet. Installed with Wi-Fi Access Points (AP) to permit mobile devices to access it, and in some cases, both of the cloudlet and AP incorporated in one system. Also, the mobile devices utilize Wi-Fi to join to the nearby cloudlet. During using the low latency, high Bandwidth and one-hop wireless access to the cloudlets mobile device gets a real-time interactive answer.

Parameters	3G / LTE	WiFi
Power consumption [1][2]	Higher	Lower
Connection Speed[3]	2Mbps	400 Mbps
Latency[3]	Higher	Lower

Table 1: 3G / LTE and WiFi comparison

2.1. Cloud-based mobile application and its execution in MCC

Cloud-based mobile applications run on mobile devices likewise on the cloud. The cloud based mobile applications composed of two types of components: transferable and non transferable. The transferable components are reminiscence compute intensive and do not relate with the mobile hardware, whereas non transferable components are intended and executed for special functionality such as hardware access, user interaction, and security related tasks [12]. The development of separating the application mechanisms into two sets, transferable and non-transferable, is called partitioning. Generally, the partitioning is executed in three different behaviours such as statically, dynamically, and semi dynamically. The mobile application execution in MCC can be illustrated by a state diagram as shown in Fig. 4. The execution of the function starts when a user clicks the application icon. The applications go into the running state where it executes different tasks. To traverse the application into the cloud, the execution framework delays the running application. The application accumulates its running states when the execution frame work motivates stays. Then the implementation control moves into the delayed state. The application and running states are moved to the cloud server where the

application is recommenced and reconfigured with the saved states. The execution control penetrates into the running state. On completion of the execution on the cloud server, the results are pressed back to the mobile device, where the application recommences its execution on the mobile device. Finally, on the achievement of the execution, application stops and enters into the terminated state.

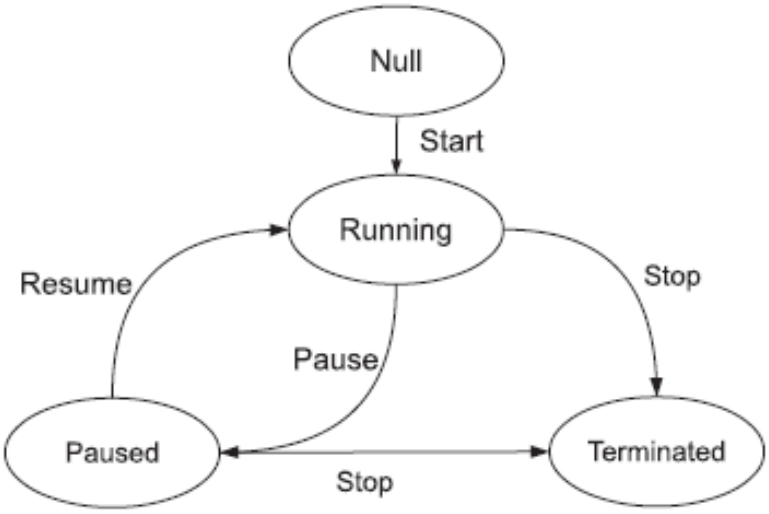

Fig. 4. Application execution in MCC

2.2. Application execution optimization

The application execution accession is an essential role for MCC because of resource forced nature of mobile devices, distributed implementation environment, and critical limitations in the communication medium. The performance of application executed in MCC improved by accomplishing the efficient design of the application, by a suitable operation, and by optimal application implementation in the scattered environment. Hence, the application execution in MCC needs efficient design, efficient service, and best execution in MCC. The practical design of an application guarantees that the application acquires low cost, higher liability, and the tremendous performance. The efficient operation guarantees that non-fiction components of the mobile use in MCC have minimal addiction on each other; thereby, sinking the operational overhead and execution cost. The optimal implementation authorizes continuous application execution with minimum user contribution. The application execution frameworks designed with the plan to amends various presentation metrics of an application. This is explained as the following Mobile Assistance Using Infrastructure (MAUI) [6] is intended to optimize the energy consumption of the mobile device.

3. UTILIZATION OF PARTITIONING ALGORITHMS IN MCC

The current utilization partitioning algorithms for expandable applications use a number of approaches for sorting out the serious components of the mobile application. This section discusses the thematic catalogue for the classification of current applications of partitioning algorithms and reviews based on the application partitioning model attribute of the taxonomy. The application partitioning algorithms are evaluated based on selected parameters presented in the taxonomy.

3.1. Taxonomy of Application Partitioning Algorithms (APA) for MCC

Figure 5 presents the particular taxonomy for the classification and comparison of APAs. The classification parameters embrace partitioning granularity, partitioning objective, partitioning model, programming language support, profiler, allocation decision, analysis technique, and annotation. The partitioning granularity aspect of an APA specifies the granularity level for partitioning computational thorough mobile application. Present APA s implements the following granularity grades for application partitioning:

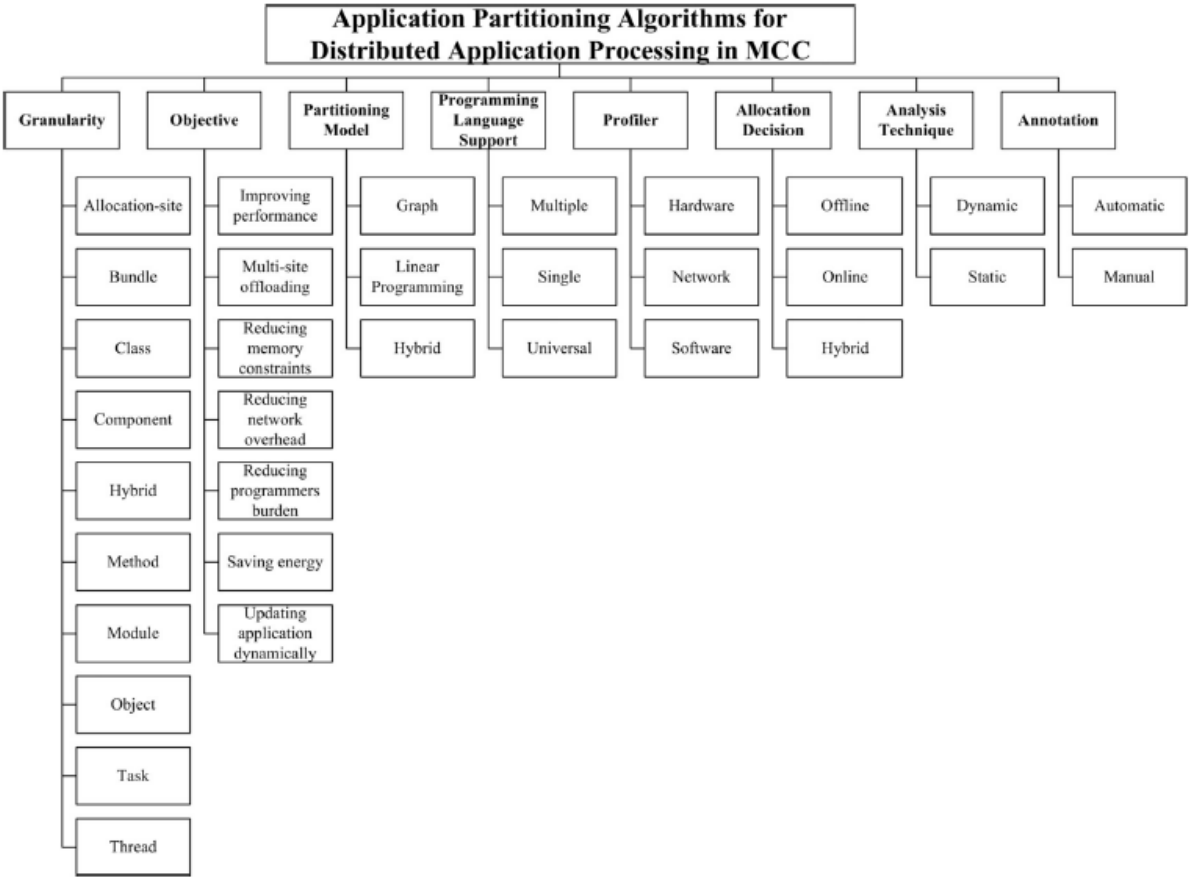

Fig. 5. Application partitioning algorithms for distributed application processing in MCC

- Module level partitioning: The whole module of the function is partitioned and scattered [13, 14].
- Method level partitioning: Partitioning develops at the method of application [6].
- Object level partitioning: The object of an application is partitioned to arrange for cyber scouring [15, 16].
- Thread level partitioning: Partitioning develops at the outfits of an application [8, 17].
- Class level partitioning: Application is partitioned in to modules for offloading [18-19].
- Task level partitioning: Application is partitioned agreeing to task [20, 21].

- Component level partitioning: Partitioning a group of modules which may or may not be joined for out sourcing to the remote server [22-28].
- Bundle level partitioning: Groups of Java grade of applications are partitioned [29-31].
- Allocation site level partitioning: Partitioning develops on the level of distribution site where all the objects at this particular site seen as a single unit [32].
- Hybrid level partitioning: The results of partitioning consist of different commonalty [33].

3.2. Review of utilization partitioning algorithms based on the application partitioning models

Present APAs categorized into three algorithms such as graph-based model, Linear Programming based model and hybrid based model.

a) Graph based application partitioning algorithm

The basics of graph, vertices and edges are used to signify the parameter of an application. These include the available resources, data size, communication, execution cost and memory cost. A mobile application modelled at different granularity levels. It includes finer granularity adaptation that marks in larger number of vertices with highly difficult interaction patterns as compared to class graphs. The suitable graph model improves the efficiency of partitioning decision in spite of whether fine grained or coarse grained granularity is functional. Fundamentally, each partition assures the condition that its weight is less than or equal to the resource accessibility of the device. A novel proposed algorithm was modelling an application as a need graph. Using this algorithm, existing applications are partitioned based on low attempt input from developers and static code analysis. Application developers are needed to explain the code units to afford signs to the partitioning algorithm. This advance describes a set of five annotations, including Mobile and Immobile annotation, to provide signs on the public cloud versus the private cloud [26].

Figure 6 (a) represents the abstract level flow chart of optimal partitioning for MCC. In graph modelling, the first step is to test whether explanation is needed. If explanation is necessary, the programmer does the explanation manually. If explanation is not necessary, the system continues to test whether the composer is available in the application. If it is available, it meets the significant information required by the application, or else, it go to next step for graph modelling. The explanations made by the programmer in addition to the profiling result are used during graph modelling. Following that, one or more algorithms are used to optimize the graph model. Finally, the result of accession is available to the deduction algorithm such as the solver to make a decision on the partitioning. In conclusion, graph based APAs minimize the input attempt with the required to make explanation. Further, graph based APAs are also computationally practical as finer level version is accessible for minimizing the connections between the element and the partition. Graph based APAs also reduce the coupling effect and movement cost for distributed application processing. These explanations, however, are less visible for maintenance. The performance of graph based APAs is severely joined to the behavior of an application, for example, whether the application is modularized or not modularized. Graph based APAs strength not recommend the best partitioning solution.

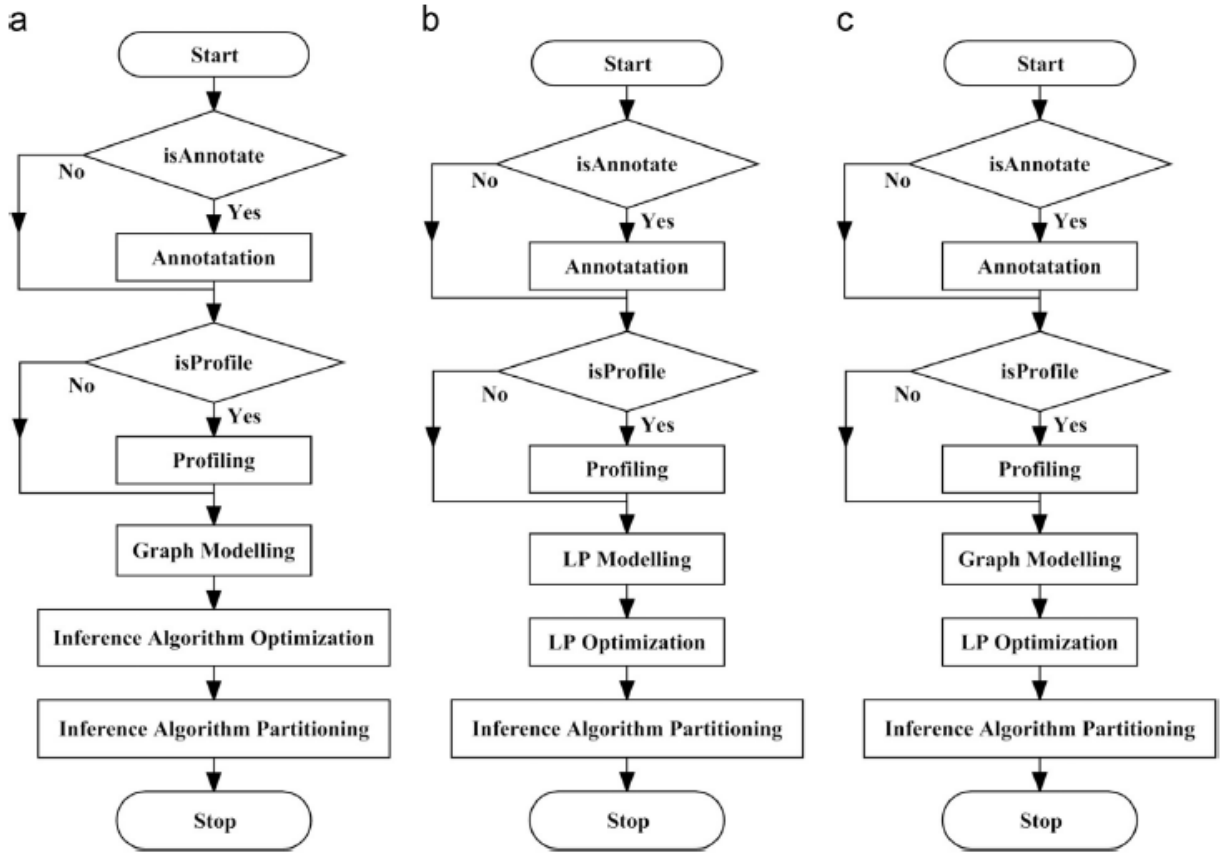

Fig. 6. A flowchart of Optimal partitioning algorithms for MCC (a) Graph-based, (b) LP based and (c) hybrid based

b) Linear programming-based application partitioning algorithms

Linear programming (LP) is a mathematical system for shaping a technique to accomplish the finest result such as maximum profit or lowest cost with a list of needs are signified as a linear equation in a mathematical model. LP is defined as a performance for the accession of a linear function; subject to line are equality and linear inequality limitations. The benefit of executing LP is that it constantly creates optimal grades for a particular objective function. LP models and develops the application as a mathematical accession problem all of the variables are controlled to be integers. Generally, LP is utilized to improve the formulated equations signify the mobile application accession problem. Figure 6(b) explains an abstract level flowchart of LP based model application partitioning. In LP modelling, the first step is to test whether explanation is needed. If explanation is needed, it appears for pre-annotated mechanism for partitioning the application. If explanation is not needed, it continues to test whether the profiler task is applied in the application. If the profiler is available, it collects the appropriate information for application partitioning. If there is not needed to profiler, it then takings to formulate the LP problem. LP supported APAs stipulate high overheads for extra profiling and resource monitoring. However, they work excellently to create the greatest and more practical partitioning

result. In addition, LP based APAs are active and lightweight to handle a large number of end users, and these also decrease the operational cost of the cloud services.

C) Hybrid application partitioning algorithms

The hybrid APAs the combined qualities from both the graph based and LP based application partitioning techniques. They attend to take out the significant features of graph based APAs and LP based APAs in order to get better the performance of APAs. Figure 6(c) describes an abstract level flowchart of hybrid based model application partitioning. In hybrid modelling, the first step is to determine whether explanation is required. If it is required, it means that the programmers have presented the terms include the explanation and information linked to partitioning. If no explanation is necessary, it goes on to test whether there is profiler in the application. If a profiler is present, significant sequence collected to take out the application partitioning. If there is no profiler, it then originates the application accession problem into a LP equation. The explanation prepared by the programmer and the profiling outcomes are useful in preparing the LP equation. LP techniques such as Integer Linear Programming (ILP), Zero One Linear Programming (0–1LP), and Mixed Integer Linear Programming (MILP), are executed to solve the formulated accession problem. Finally, the optimization result is presented to the deduction algorithm to make a result on the partitioning. The mobile device controls at least the user facing modules such as the user interface, and one or more servers are used for computation offloading or for co-locate computation with data for reduce communication costs. But, hybrid APAs are aimed at single exact type of computation system, for example, freely coupled multiprocessor system and data stream system.

4. SECURITY CHALLENGES IN MCC

The next significant computing subject is the security. The success of MCC is founded on how users' data and appliances are secured from unauthorized user. There are two kinds of security in MCC, security for mobile user and security for data appliances. Mobile User Securities are two different issues in the mobile user security. The first one is frequent security threats, malwares and adware attack on mobile devices. The second one is installation of GPS in mobile devices directs to privacy issues. These two problems addressed by installing security, malware and adware alerts up to some extent. Privacy Location Based Service (LBS) develop into a privacy issue in MCC when users distribute private information like location and other private information.

Data Security on Cloud Mobile users accumulate large amount of data on the cloud and they should be careful in dealing with the data. There are three kinds of issues look upon to data security. The digital right is very essential in this major period of progressing technologies. The data is outsourced to cloud servers to expand the storage capacity as the mobile users mislay the physical control of their data simultaneously. Thus, the accuracy of the data happens to one of the concerns for mobile users in cloud storage development. Although the cloud infrastructures are much greater consistent and powerful than mobile devices, they are still confronting a multitude of threats from internal and external for data integrity.

5. OPEN CHALLENGES IN MCC

In this section, we highlighted some of the most critical challenges that obstruct the optimization of mobile application, application partitioning algorithms and security challenges. The designing of most favourable application and implementation framework present the best application execution for such different and dynamic environment is an open research challenge. The efficient deployment of application involves minimal dependency cross non-collocated components, reducing the distance to the available resources, minimal bandwidth utilization, and low runtime cost. The real-time execution of the heterogeneous computing environment happens to a challenge due to diverse nature of environmental elements, intrinsic limitations of the wireless medium, and the highly dynamic mobility patterns. It is also reviewed the APA s in distributed application processing to recognize the main issues and challenges. Present application partitioning techniques characterized based on thematic taxonomy and other vital aspects evaluated. Current APAs agree to different models for modelling elastic applications such as graph based model, the LP optimization problem for application partitioning and the hybrid modelling. Some algorithms are limited to the use of a single programming language, where as others use multiprogramming language support. APAs incorporate different types of profilers, which include network, software, and hardware profilers. The present APAs focussed on separating the demanding components of the application as profiling the accessibility of resources on local mobile device and do not think the availability of services on the computational clouds. The computational offloading procedures do not add in an error understanding mechanism to deal with partition loss. Also, the active application shape mechanism in application partitioning is time-consuming, energy consuming and resources demanding. Arising from signals is at risk to hindrance and concerning, a mobile computing network system measured from terminals, networks, database platforms, further applications improvement to deal with the security issue. The mobile cloud securities are faced in three main points. 1) To maintain the information stored in the cloud secure resources the data stored by the user might not be admittance by the third party. The data stored in encoded form in the cloud but that data is also identified to the cloud provider in a decoded form as encoding executed by the cloud providers only, but they should be trustworthy. 2) The implementation of the attractive code which doesn't enclose any virus or malware in the system only that code should implement at the time of improvement of the code some malicious content initiated. This software should be revalidated to preserve guaranteed validity. Problem occurs when stream of attainable set up quickly spread malicious payload on cloud computing environment. 3) The information stored and services afforded by the cloud to the user at all times are available when needed. For the reproduction of data is completed it directs to problem of data management to guarantee regularity. All- time the services modernized can address failure. The information saved on the cloud by the user through mobile devices should integrate. Users know where the data is reserved and who can right to use it and every access can be approved to keep the confidential information can do.

6. CONCLUSIONS

This paper investigated the mobile cloud computing application, optimization strategies, application partitioning algorithms and security challenges by surveying the literature. The present function implementation frameworks develop the reply time of application for MCC by dropping the massive area network time delay. Many current application implementation frameworks need with scalability features and QoS support from profiling overhead. Most of the literature survey discussed security issues look into clouds any changes providing the difference between susceptibility and threats. We have focused on this direction, the data security essential to know these problems. These security issues were not enough; this idea we produce a communication among threats and vulnerabilities. We can found what weaknesses contribute to the execution of these risks and make the system more robust. The current solution solves to moderate these threats. In spite of, new security techniques required and further especially standard solutions can task with cloud architectures. Traditional security structures may not work well in cloud environments. Thus it is a future research work focus on different design architecture that is confident of a combination of different technologies.

REFERENCES

[1]. S. Gunelius, "Mobile Devices to Surpass the Number of People on Earth" [Online]. Available: http://newstex.com/2014/05/03/mobile devices to surpass the number of people on earth in fographic, May 2014.

[2]. J. Flinn, S. Park, and M. Satyanarayanan, "Balancing performance, energy, and quality in pervasive computing," Proceedings of the 22nd International Conference on Distributed Computing Systems, pp. 217-226, Nov 2002.

[3]. R. Balan, M. Satyanarayanan, S. Park, and T. Okoshi, "Tactics-based remote execution for mobile computing," Proceedings of the 1st ACM International Conference on Mobile Systems, Applications and Services, pp. 273-286, May 2003.

[4]. E. E. Marinelli, "Hyrax: cloud computing on mobile devices using Map Reduce," Masters Thesis, Carnegie Mellon University, 2009.

[5]. R. Iyer, S. Srinivasan, O. Tickoo, Z. Fang, R. Illikkal, S. Zhang, V. Chadha, P. Stillwell, and S. Lee, "Cogniserve: Heterogeneous server architecture for large scale recognition," *IEEE Micro*, vol. 31, no. 3, pp. 20-31, April 2011.

[6]. E. Cuervo, A. Balasubramanian, D. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "Maui: making smartphones last longer with code offload," Proc. 8th ACM international conference on Mobile systems, applications, and services, pp.49-62, June 2010.

[7]. M. Satyanarayanan, P. Bahl, R. Caceres, N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14-23, Oct 2009.

[8]. B. G. Chun, S. Ihm, P. Maniatis, M. Naik, A. Patti, "Clone cloud: elastic execution etween mobile device and cloud," Euro Sys '11, Proceedings of the sixth conference on Computer systems, pp. 301-314, April 2011.

[9]. D. Huang, X. Zhang, M. Kang, J. Luo, "Mobicloud: building secure cloud frame work for mobile computing and communication," Fifth IEEE International Symposium on Service Oriented System Engineering (SOSE), pp. 27-34, Sep 2010.

[10]. S. Kosta, A. Aucinas, P. Hui, R. Mortier, X. Zhang, "ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," INFOCOM, pp. 945-953, May 2012.

[11]. S. Kim, Y. Yonglk, Minkyo In, L. Kangchan, L. Seungyun, "The evolution of standardization for mobile cloud," International Conference on ICT Convergence (ICTC), Nov 2011.

[12]. R. K. Balan, D. Gergle, M. Satyanarayanan, J. Herbsleb, "Simplifying cyber foraging for mobile devices," In: Proceedings of the fifth international conference on mobile systems, applications, and services (MobiSys'07),SanJuan,PuertoRico.ACM, pp.272-285, June 2007.

[13]. L. Yang, J. Cao J, H. Cheng, Resource constrained multiuser computation partitioning for interactive mobile cloud applications. Technical report. Department of Computing, Hong Kong Polytechnical University 2012.

[14]. E. Tilevich, Y. Smaragdakis, J-Orchestra automatic java application partitioning. In: Proceedings of the16th European conference on object oriented programming (ECOOP'02), Malaga, Spain.Springer-Verlag, pp.178-204. May 2002.

[15]. J. Niu, W. Song, M. Atiquzzaman, "Bandwidth adaptive partitioning for distributed execution optimization of mobile applications," *Journal of Network and Computer Applications*, vol. 37, pp.334-347. Jan 2014.

[16]. R. Newton, S. Toledo, L. Girod, H. Balakrishnan, S. Madden, Wishbone: Profile-based partitioning for sensor net applications. In: Proceedings of the sixth USENIX Symposium on networked systems design and implementation (NSDI'09), Boston, MA.USENIX Association Berkeley; pp. 395-408. Apr 2009.

[17]. E. Abebe, C. Ryan, "Adaptive application offloading using distributed abstract class graphs in mobile environments," *Journal of Systems and Software*, vol. 85, no. 12, pp. 2755-2769, Dec 2012.

[18]. X. Gu, K. Nahrstedt, A. Messer, I. Greenberg, D. Milojicic, Adaptive offloading inference for delivering applications in pervasive computing environments. In: Proceedings of the first IEEE international conference on pervasive computing and communications (PerCom'03),Lugano, Switzerland. IEEE, pp.107–14, April 2003.

[19]. S. Ou, K. Yang, A. Liotta, An adaptive multi constraint partitioning algorithm for offloading in pervasive systems. In: Proceedings of the fourth annual IEEE international conference on pervasive computing and communications (Per- Com'06), Pisa, Italy. IEEE; pp. 116-125, Mar 2006.

[20]. C. Wang, Z. Li, "Parametric analysis for adaptive computation offloading," Proceedings of the ACM SIGPLAN 2004 conference on Programming language design and implementation, pp. 119-130, June 2004.

[21]. M. Goraczko, J. Liu, D. Lymberopoulos, S. Matic, B. Priyantha, F. Zhao, Energy optimal software partitioning in heterogeneous multi processor embedded systems. In: Proceedings of the 45th annual design automation conference (DAC'08), Anaheim, CA, USA. ACM, pp. 191-196, June 2008.

[22]. R. Bialek, E. Jul, J. G. Schneider, Y. Jin, Partitioning of java applications to support dynamic updates.In:Proceedingsofthe11thAsia-Pacific software engineering conference (APSEC2004),Busan, Korea. IEEE, pp. 616-623, Dec 2004.

[23]. H. Chu, H. Song, C. Wong, S. Kurakake, M. Katagiri, "Roam a seamless application framework," *Journal of Systems and Software*, vol. 69, no. 3, pp. 209-226, Jan 2004.

[24]. L.D. Pedrosa, N.Kothari, R. Govindan, J. Vaughan, T. Millstein, The case for complexity prediction in automatic partitioning of cloud-enabled mobile applications. Technical report. Los Angeles, USA: University of Southern California; June 2012.

[25]. M.R. Ra, B. Priyantha, A. KansalA, J. Liu, Improving energy efficiency of personal sensing applications with heterogeneous multiprocessors.In:Proceedingsofthe14th international conference on ubiquitous computing (Ubicomp'12), Pittsburgh, PA,USA.ACM, pp. 1-10, Sep 2012.

[26]. M. Smit, M. Shtern, B. Simmons, M. Litoiu, Partitioning applications for hybrid d and federated clouds. In: Proceedings of the conference of the center for advanced studies on collaborative research (CASCON'12),Toronto, Ontario, Canada.IBM Corp, pp. 27-41. Nov 2012.

[27]. L. Yang, J. Cao, S. Tang, T. Li, A.T.S. Chan, A framework for partitioning and execution of data stream applications in mobile cloud computing. In: Proceedings of the IEEE fifth international conference on cloud computing (CLOUD'12),Honolulu, HI, USA.IEEE, pp. 794-802. Aug 2012.

[28]. T. Verbelen, T. Stevens, F. D. Turck, B. Dhoedt, "Graph partitioning algorithms for optimizing software deployment in mobile cloud computing," *Future Generation Computer Systems*, vol. 29, no. 2, pp.451–459, Feb 2013.

[29]. I. Giurgiu, O. Riva, D. Juric, I. Krivulev, G. Alonso, Calling the cloud: enabling mobile phones as interfaces to cloud applications. In: Proceedings of the ACM/IFIP/ USENIX 10th international conference on middleware (Middleware'09), Urbana, IL,USA. Springer-Verlag, pp. 83-102, 2009.

[30]. I. Giurgiu, O. Riva, G. Alonso, Dynamic software deployment from clouds to mobile devices. In: Proceedings of the 13th international middleware conference (Middleware'12), Montreal, QC, Canada. Springer-Verlag, pp. 394-414. Dec 2012.

[31]. D. Kovachev, R. Klamma, "Framework for computation offloading in mobile cloud computing," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 1, no. 7,pp. 6-15, Dec 2012.

[32]. K. Sinha, M. Kulkarni, Techniques for fine grained, multisite computation offloading. In: Proceedings of the 11th IEEE/ACM international symposium on cluster, cloud and grid computing (CCGrid'11), Newport Beach, CA, USA.IEEE, pp. 184-194. May 2011.

[33]. E. Abebe, C. Ryan, A hybrid granularity graph for improving adaptive application partitioning efficacy in mobile computing environments. In: Proceedings of the 10th IEEE international symposium on network computing and applications (NCA'11), Cambridge, MA,USA. IEEE, pp. 59-66. Oct 2011.

