International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 1 | ISSN : 2456-3307

Survey on Mobile Cloud Computing : Applications, Techniques and Issues

Monika Sharma[1], Raj Kumari[2]
University Institute of Engineering and Technology Panjab University, Chandigarh, India

ABSTRACT

The mobile technology becomes robust in the present scenario because now days, mobile devices become capable to support huge variety of applications. On the second thought, the mobile devices are experiencing many challenges as they have narrow pool of resources like battery life, bandwidth and capacity for storing data etc. But after the emergence of mobile cloud computing technologies, the mobile services become more prominent. Mobile Cloud Computing (MCC) refers to cluster of cloud computing and mobile networks where mobile applications transfer the computation power and storage from the mobile devices to the cloud to enhance the capabilities of mobile devices. In MCC, computation offloading is a technique that helps in transferring the complex application modules that require intense computation from a mobile device to the resource-rich cloud. The benefits of cloud based computation offloading are that it helps in improving the performance of integrated application, also enhances the overall execution time of the application and improves battery life of mobile devices. This paper focuses on offloading benefits, its techniques, its related issues and challenges.

Keywords: Cloud Computing, Mobile Cloud Computing, Computation Offloading.

I. INTRODUCTION

Before the concept of MCC, the technology of Cloud Computing was introduced in 1996. Cloud Computing is an emerging computing technology that preserves data and applications by using the internet and central remote servers. It helps in configuring and customizing the applications efficiently. With Cloud Computing users can access the resources through internet from anywhere, at any time with pre-defined charges which is known as "pay-as-you-use principle."

In order to take the advantage of Cloud Computing, an abstract idea of MCC was introduced. MCC is the concept that combines the Cloud Computing and the mobile networks. In MCC, the data computation as well as data storage takes place in remote cloud servers, outside of the mobile device. This is because mobile devices are resource-constrained devices; they have fewer resources (i.e. battery life, network bandwidth and storage capacity). MCC provides mobile device users with data storage and processing services in clouds, as all the resource intensive computing can be carried out in cloud. Therefore, there is less consumption of mobile device resources and hence making mobile devices more efficient.

1.1 The Architecture of MCC

MCC architecture is as follows: Multiple mobile devices are linked to the mobile networks through base stations such as BTS, satellite or access points.

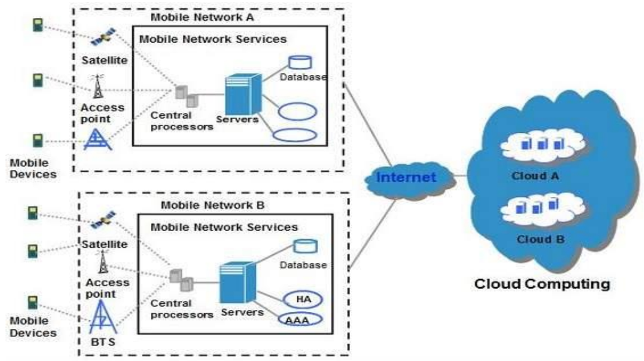

Figure 1. MCC Architecture [1]

These are capable to create and control the connections and functional interfaces between the networks and mobile devices. Mobile user's information and requests are transmitted to the central processors that are connected to servers, providing mobile network services. Now the users are given the services (like authentication, authorization, accounting based on data stored in database) by the mobile network operator. After authorizing user, the cloud server receives the users request with the help of internet. Basically, internet plays the role of intermediator that sends the user requests from mobile networks to the cloud. Afterwards, cloud controllers present in the cloud are responsible to process the user requests and provide the corresponding cloud services.

1.2. Applications In MCC:

Mobile applications have been used on large scale and have a great share in a worldwide mobile market. Various mobile applications like mobile commerce, mobile learning, mobile sensing, mobile healthcare, mobile gaming, multimedia sharing, mobile social networking and many more; have taken advantage of mobile cloud computing. In this section, some of the typical MCC applications are discussed briefly.

1.2.1. Mobile Gaming

Mobile games expected to have small scope as mobile devices lack high processing power, which is required for the graphic rendering. As a result, it generally depends on simple play rather than graphics. Due to this, mobile gaming use to transfer the game engine to the cloud as it is computation intensive and local device is not able to handle such application modules that needs high computing power and cloud in return sends the output after computing which the game users receive on their mobile devices. This standard brings many benefits such as power saving, enhancing game playing speed due to cloud's high computation speed.

MAUI (memory arithmetic unit and interface), is a system that emphasizes more on energy saving, so at the run time, it divides the application codes based on the cost of the network communication and CPU processing speed on the local mobile device. The results show it helps in energy reduction remarkably for the mobile device. Moreover, the performance of mobile applications is also improved. The diagnosed statistics shown by MAUI are: energy saved for video game is 27% and 45% for chess while the refresh rate of game maximizes from 6 to 13 frames per second.

1.2.2. Mobile Learning

Mobile learning applications enable users to have distance learning via mobile gadgets such as tablets, notebooks, mobile phones etc. These m-learning applications are basically the electronic learning with mobility. It provides m-learners flexibility, as they can easily access the applications from anywhere, at any time, from any portable device and in any format. In comparison to traditional m-learning applications, Cloud-based m-learning applications provide users distance education at faster processing speed with more educational resources at high network transmission rate.

The interaction status between the students and teachers has increased with the evolution of m-learning when combined with cloud computing. Through a web site built on Google Apps Engine, students [9] can interact with their teachers at any time. Also, the information regarding the student's understanding level of the course can be drawn by the teachers and can provide solutions to student's queries in a timely manner.

1.2.3. Mobile Healthcare

Mobile healthcare applications are medical applications in the mobile environment used for diagnosing the health issues, helps in providing medical treatment etc. In tradition medical applications have the limitations such as less storing capacity, minimal security, and high rate of medical errors. But, these limitations are diminished after applying the MCC concept in health applications. Mobile healthcare applications enable users to access the resources at faster pace at any time, from any place. With the use of cloud, these applications provide variety of on-demand services on clouds. Using such mobile healthcare applications, there arise challenges to safeguard the user's health information, so there have to be proposed solutions to increase the privacy of the users, as have to be done in the traditional applications.

Therefore, a solution for protecting participant's health information is proposed by [10]. P2P model has been used to enhance the privacy of the services provided to mobile users. This model associates the clouds to report security related issues and safeguards the data.

1.2.4. Mobile Commerce:

Mobile commerce is the capability that provides commerce using a mobile device. The m-commerce applications handle multiple tasks that need mobility functions such as mobile banking, mobile money transfer, mobile marketing, mobile browsing, mobile purchasing etc. M-commerce related services are increasing rapidly, due to this there arouse various challenges such as low network bandwidth, and security related problems.

Therefore, 3G E-commerce platform based on cloud computing [14] mingles the benefits of both cloud computing and 3G network to enhance the speed at which the data is being processed and security level [15] based on PKI (public key infrastructure), which operates on an encryption-based access mechanism. It helps in safeguarding the subscriber's privacy and it also provides secure access to data stored in cloud.

II. CONCEPT OF OFFLOADING IN MCC

Offloading is a method of migrating the application modules, that require complex processing and heavy computations, from the local mobile devices (i.e. resource-constrained devices) to remote cloud servers (i.e. resource rich devices).

The concept of MCC involves the offloading of the task that is to be executed by the remote server. The part of application that need to be offloaded from mobile phone to cloud could be done in two fashions that is partial offloading or full offloading [27,28]. In the full offloading architecture the full application along with all the data associated to it has been offloaded to the cloud where the entire computation take place and the final results have been sent back to the mobile device as shown in figure 2.

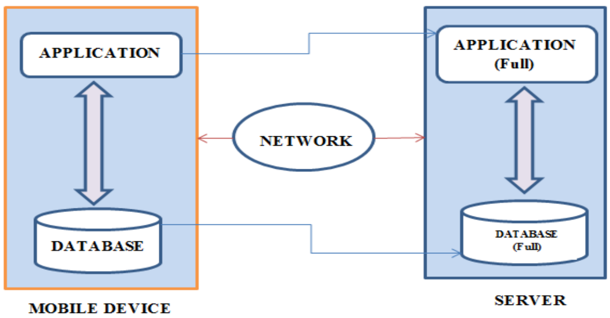

Figure 2. Full Offloading

In the partial offloading architecture, only that part of the application which consumes more energy or have high complexity in terms of computation has been offloaded to the cloud. In this, both mobile phone and the cloud are responsible for the computation and final results comes after merging the individual results of both the computations that is in mobile device and at the cloud as shown in fig.3.

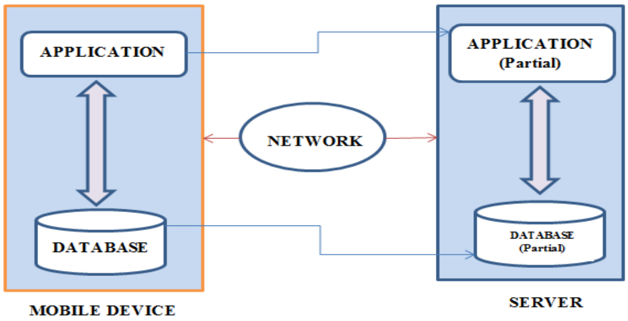

Figure 3. Partial Offloading

- Offloading decision helps to: (i) improve performance, (ii) save energy, (iii) Increase storage capacity, (iv) Increasing reliability

Conditions for improving performance and saving energy through offloading are presented [19]. These conditions are as follows:

W	Amount of computational data
S(m)	Speed of the mobile system
D(i)	Sending input data
T(m)	Total Time to execute the application in mobile device
S(c)	Server Speed
T(c)	Total Time to offload and execute application in cloud
B	Bandwidth
P(m)	Power on the mobile system
P(c)	Power required to send data from the mobile system over the network
P(i)	Power consumption during execution of data in server

1. **Improving Performance:** The condition for offloading to upgrade performance is as discussed:

In mobile device,

$$T(m): \frac{W}{S(m)} \quad (1)$$

In cloud,

Time taken to send input data over bandwidth B, T_1: $\frac{D(i)}{B}$

$$T(c): T_1 + \frac{W}{S(c)} \quad (2)$$

Offloading upgrades performance when Eq. 1 > Eq. 2

$$T(m) > T(c) \quad (3)$$

2. **Saving Energy:** The major limitation in mobile devices is energy, which needs to be alleviated by offloading. Energy saving during offloading can be analysed by the following condition.

In mobile device,

The energy required to perform the task can be attained by modifying Eq. 1:

$$Energy(m): Pw(m) \times \frac{W}{S(m)}$$

In cloud,

The energy required to perform the task can be attained by modifying Eq. 2:

$$Energy(c): P(c) \times \frac{D(i)}{B} + P(i) \times \frac{W}{S(c)} \quad (4)$$

Energy is saved through offloading when Eq. 3 > Eq. 4

Energy(m) > Energy(c)

In both equations, we assume that data must be transmitted from the mobile system to the server. Portable computers [18] that offload their heavy (i.e. require more computational power) tasks to the remote servers in order to save the battery power. It was observed that approximately 51% energy is saved. On the basis of power consumed by the network subsystem, a new routing method [19] is introduced. This method demonstrates the enhancement in power consumption i.e. up to 15% on an average and reduction in latency over 75%. This method shows better results as compared to the methods that rely only on the transmitted power.

3. **Increase Reliability:** When the data is stored in cloud, it is always ensured that the data is protected and backed up on multiple devices. This minimizes the possibility of data loss even in the adverse situations like power failure or other crisis. Therefore, a new scheme, that ensure the license's integrity [20] and that can protect licenses from being misused and illegally modified. Moreover, some security services are remotely offered by the cloud to its users. Such services include virus scanning and malicious code detection [21].

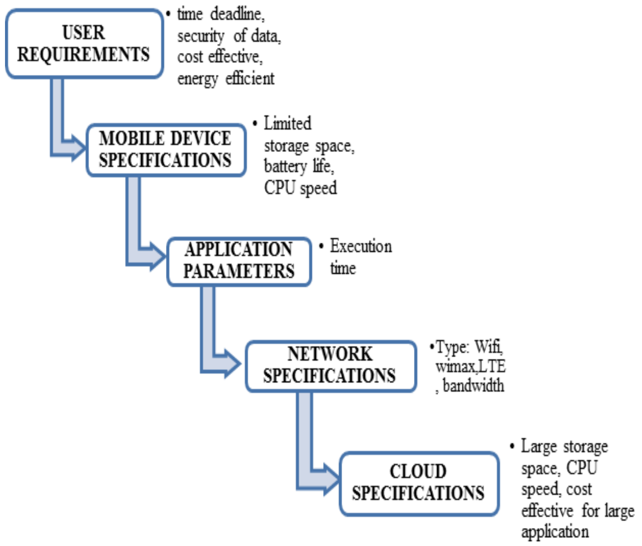

Figure 4. Aspects for Offloading Decision

But it is crucial to decide if to offload or not, and which module of the application require to be offloaded. The offloading decision depends upon the following aspects as shown in Fig.4.

Offloading decisions are based upon the following aspects: User Requirements, Mobile Device Specifications, Application Parameters, Network Specifications and Cloud Specifications

2.1 Techniques for Offloading of an Application:

2.1.1. Static Offloading is easier to implement as the application is partitioned into modules during the early phase of development. In static environment, criterions like size of the data, energy consumption, and execution time are known beforehand, however it may not yield the promised offloading benefits. The related work on static offloading has been discussed further.

On the basis of estimation of energy consumption, program partitioning takes place before the program execution. Through calculating trade-off between the communication and computation costs, the optimal program partitioning for offloading [3] is calculated. But, offloading scheme [4] based on describing information about computation time and data sharing at level of procedure calls is described, where the cost graph is constructed and if we apply the branch and bound algorithm on it, then this result in minimization of total energy consumption.

Before the program implementation, on basis of estimation of the energy expenditure, program splitting is done. On the basis of trade-off between the calculation and transmission rates, the optimum program splitting for offloading is computed. Some explanations are supported to find the optimal result for partitioning functions before offloading.

For some offloading methods, the prior estimation of execution time is not required. For such methods online statics of computation time [2] are used in order to calculate the optimal timeout, and if the computation is not completed within time limit, then it is migrated to remote servers. It results in saving up to 17% energy as compared to the other methods. An automatic distributed partitioning system (ADPS) called Coign, [26] demonstrates the optimal distribution of program that is automatically converted to the distributed application. In this, conversion takes place without using the source code of a program and a graph model is constructed to observe the distribution.

2.1.2. Dynamic Offloading is harder to implement as in dynamic environment the modules of an application may be transferred to cloud when the application is running, however it may results in higher performance and may yield maximum benefits since offloading is performed only when necessary.

Offloading systems in wireless environments [5] is analysed, this examines three circumstances in order to evaluate the fact that how efficient the offloading is when we execute the application on cloud servers. Firstly, the whole application is executed locally on mobile device. In this, no offloading takes place. Secondly, the offloading is performed but without considering any failure. In the third aspect, the offloading takes place but with the failure recoveries.

The sub-modules that failed to offload, they are re-offloaded after failure.

A system follows three steps for partitioning [6] an application. The first step is the application structuring. In this, the application run dynamically between the mobile and cloud. At the run time, the application decides which module should run at the cloud server and which part of the application to be executed locally. Second step involves the partitioning choice. In order to minimize the energy consumption, the system applies the optimal strategy. Third step is the security. Rather than offloading the application module containing sensitive data, it is executed locally on mobile device.

The decision between the local and remote executions can be made by dynamically evaluating [22] the trade-offs between the computational cost and communication cost. This evaluation is based on input parameters and channel conditions.

Dynamic performance optimization framework for MCC using mobile agent based application partitions [24]. Application partitioner component partition the mobile application during its installation on the device. When the application is launched by the user, the virtual machine instances are located to the offloaded application modules. This allocation of instances is done by the offloading manager component of the framework as it contacts the cloud registry. Then the offloaded modules of an application are sent over the network to their respective instances for execution. Then the whole application is executed with the collaboration of sub-modules. Then these sub-modules are integrated.

III. ISSUES AND CHALLENGES IN OFFLOADING

3.1. Security: Security is the foremost issue of the users in mobile environment. The major concern is the protection of private data of users. For instance, the sensitive data of the user stored in the cloud can be misused by the cloud service provider, without the awareness of the user. For example, when visualizing the integrated global positioning system (GPS) devices [12], the privacy concern arises. The GPS devices are considered to be more security intensive because it can cause subscribers to be tracked. Hence, it is a major concern to provide security and privacy to users.

3.2. Authentication: As huge amount of data/applications are stored on a cloud both the cloud providers and the users should be careful while tackling with the sensitive data or applications. Such sensitive data need to be authenticated to avoid any misuse. Users should have authorised keys [13] while using external resources, therefore secure authentication mechanisms should be implemented.

3.3. Low Bandwidth: Bandwidth is the critical issue in mobile cloud environment. Bandwidth utilisation [16] relies upon the sub modules of the application that are offloaded from local mobile device to the remote cloud. If the offloaded data is huge then it results in the delay between the transferring the data on cloud and the final result to be sent back to local mobile device which in turn results in less efficiency and high energy consumption.

3.4. Heterogeneity: In mobile environment, WCDMA, GPRS, WiMAX, and WLAN are various network types that are used simultaneously for mobile communication. As a result, it becomes very difficult to handle heterogeneous network [13] connectivity while fulfilling mobile cloud computing requirements (i.e. scalability, improved mobility, on-demand availability etc.). So, heterogeneity is a major problem in computational offloading.

IV. CONCLUSION

MCC is one of the emerging mobile technologies as it collaborate the benefits of both mobile computing and cloud computing, thereby delivering best services for mobile users. We discussed few mobile applications that are taking advantage of MCC because mobile devices do not encourage some complex applications to be computed locally as these devices face deficiency of enough hardware, software and battery lifetime. Offloading is technique of MCC that migrates the heavy computations to cloud servers which are more resourceful and the results are received by the mobile systems. This method enables the mobile device users to access all those applications which require large memory storage, high computational power, long battery lifetime and large network bandwidth. In this paper, we analyse the situation in which performance can be improved and energy can be saved by offloading. We listed down some of the research areas related to offloading. Further this paper presented some major issues in offloading such as security, authentication, low bandwidth, heterogeneity. These issues eventually devastate the efficiency of the process. So, these issues should be taken into consideration for the future work.

V. REFERENCES

[1]. R.N. Calherios, R. Ranjan, A. Beloglazv, C.A.F.D. Rose and R. Buyya,"CloudSim:a toolkit for modelling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," in Software-Practice and Experience, vol.41, pp.23-50, 2011.

[2]. C. Xian, Y. H. Lu, and Z. Li, "Adaptive computation offloading for energy conservation on battery-powered systems" in Intl Conf on Parallel and Distributed Systems, vol. 2, pp. 1-8, December 2009.

[3]. K. Kumar and Y. Lu, "Cloud Computing for Mobile Users: Can Offloading Computation Save Energy," IEEE Computer, vol. 43, no. 4, April 2010.

[4]. Z. Li, C. Wang, and R. Xu, "Computation offloading to save energy on handheld devices: a partition scheme" in Proceedings of 2001 Intl Conf on Compilers, architecture, and synthesis for embedded systems (CASES), pp. 238-246, Nov 2001.

[5]. S. Ou, K. Yang, A. Liotta and L. Hu, "Performance Analysis of Offloading Systems in Mobile Wireless Environments" 2007 IEEE International Conference on Communications, Glasgow, pp. 1821-1826, August 2007.

[6]. BG. Chun and P. Maniatis, "Dynamically partitioning applications between weak devices and clouds" in Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond (MCS), no. 7, June 2010.

[7]. G. Chen, B. T. Kang, M. Kandemir, N. Vijaykrishnan, M. J. Irwin and R. Chandramouli, "Studying energy trade offs in offloading computation/compilation in Java-enabled mobile devices" in IEEE Transactions on Parallel and Distributed Systems, vol. 15, no. 9, pp. 795-809, Sept. 2004.

[8]. E. Cuervo, A. Balasubramanian, D. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "MAUI: Making Smartphones Last Longer with Code offload," in Proceedings of the 8th International Conference on Mobile systems, applications, and services, pp. 49-62,June 2010.

[9]. W. Zhao, Y. Sun, and L. Dai, "Improving computer basis teaching through mobile communication and cloud computing technology," in Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), vol. 1, pp. 452-454,September 2010.

[10]. D. B. Hoang, and L. Chen, "Mobile Cloud for Assistive Healthcare (MoCAsH)," in Proceedings of the 2010 IEEE Asia-Pacific Services Computing Conference (APSCC), pp. 325-332, February 2011.

[11]. K. Akherfi, M. Gerndt, and H. Harroud, "Mobile cloud computing for computation offloading: Issues and challenges," in Applied Computing and Informatics, 2016.

[12]. Z. Zhang and S. Li, "A Survey of Computational Offloading in Mobile Cloud Computing," 2016

4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), Oxford, 2016, pp. 81-82.

[13]. H. T. Dinh, C. Lee, D. Niyato, P. Wang, "A survey of mobile cloud computing: architecture applications and approaches," in Wireless Communications and Mobile Computing, 2011.

[14]. X. Yang, T. Pan, and J. Shen, "On 3G Mobile E-commerce Platform Based on Cloud Computing," in Proceedings of the 3rd IEEE International Conference on Ubi-Media Computing (U-Media), pp. 198-201, August 2010.

[15]. J. Dai, and Q. Zhou, "A PKI-based mechanism for secure and efficient access to outsourced data," in Proceedings of the 2nd International Conference on Networking and Digital Society (ICNDS), vol. 1, pp. 640, June 2010.

[16]. Roopali, Rajkumari, "Overview of Offloading in Smart Mobile Devices for Mobile Cloud Computing," in International Journal of Computer Science and Information Technologies, vol. 5(6), 7855-7860, 2014.

[17]. G. Chen, B. T. Kang, M. Kandermir, N. Vijaykrishnan, M. J. Irwin, and R. Chandranouli, "Studying energy trade offs in offloading computation/compilation in Java-enabled mobile devices," in IEEE Transactions on Parallel and Distributed Systems, vol. 15, no. 9, pp. 795, September 2004.

[18]. A. Rudenko, P. Reiher, G. J. Popek, and G. H. Kuenning, "Saving portable computer battery power through remote process execution," in Journal of ACM SIGMOBILE on Mobile Computing and Communications Review, vol. 2, no. 1, January 1998.

[19]. A. Smailagic and M. Ettus, "System Design and Power Optimization for Mobile Computers," in Proceedings of IEEE Computer Society Annual Symposium on VLSI, pp. 10, August 2002.

[20]. P. Zou, C. Wang, Z. Liu, and D. Bao, "Phosphor: A Cloud Based DRM Scheme with Sim Card," in Proceedings of the 12th International Asia-Pacific on Web Conference (APWEB), pp. 459, June 2010.

[21]. J. Oberheide, K. Veeraraghavan, E. Cooke, J. Flinn, and F. Jahanian, "Virtualized in-cloud security services for mobile devices," in Proceedings of the 1st Workshop on Virtualization in Mobile Computing (MobiVirt), pp. 31-35, June 2008.

[22]. G. Chen, B. T. Kang, M. Kandemir, N. Vijaykrishnan, M. J. Irwin and R. Chandramouli, "Studying energy trade offs in offloading computation/compilation in Java-enabled mobile devices," in IEEE Transactions on Parallel and Distributed Systems, vol. 15, no. 9, pp. 795-809, Sept. 2004.

[23]. K. Kumar and Y. H. Lu, "Cloud Computing for Mobile Users: Can Offloading Computation Save Energy?," in Computer, vol. 43, no. 4, pp. 51-56, April 2010.

[24]. P. Angin, B. Bhargava. "An Agent-based optimization framework for mobile-cloud computing," in Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, vol. 4, no. 2, 2013.

[25]. X. Zhang, J. Schiffman, S. Gibbs, A. Kunjithapa, and S. Jeong, "Securing elastic applications on mobile devices for cloud computing," in Proceedings of the ACM Workshop on Cloud Computing Security (CCSW'09), Chicago, Illinois, USA. ACM, pp. 127–134, November 2009.

[26]. G. C. Hunt and M. L. Scott, "The Coign automatic distributed partitioning system," in Proc 3rd Symposium on Operating systems design and implementation (OSDI), pp. 187-200, Feb 1999.

[27]. J. Kim, "Design and evaluation of mobile applications with full and partial offloadings." Advances in Grid and Pervasive Computing, Springer Berlin Heidelberg, pp. 172-182, 2012.

[28]. M. Shiraz and A. Gani, "A lightweight active service migration framework for computational offloading in mobile cloud computing" in The Journal of Supercomputing, pp. 1-18, 2014.