International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 11 (2017) pp. 2754-2760

Improving the QoS of Mobile Cloud Computing Applications in Smart Environments

C.T. Meenatchi Sundaram[1] and Dr. A.R. Mohammed Shanavas[2]

[1]Research Scholar, Computer Application Department, PRIST University, Vallam, Tanjavur, Tamil Nadu- 613403, India.

[2]Associate Professor, Department of Computer Science, Jamal Mohamed College, P.Box.No.808,7, Race Course Road, Khajanagar,Tiruchirapalli-620 020, Tamil Nadu, India.

[1]ORCID: 0000-0001-5768-5715

Abstract

Mobile-edge cloud computing is a new computing paradigm which provides cloud computing facilities to the mobile users in the close proximities of the edge of pervasive radio access networks. This paper proposes a novel approach to enhance the QoS of multimedia streaming in the context of smart environments. In the proposed scheme Mobile Edge Computing plays a vital role in enhancing QoS by a new method called the Track the Edge Approach. We have considered the applications of mobile cloud for developing the smart environments. We explore the container-based virtualization techniques to provide active Mobile Edge Computing (MEC) environments. In particular, our scheme guarantees fast response time, by proactively increasing service replication. We have created a test bed for conducting experiments and the results prove that the significant improvements in the performance of the proposed strategy over traditional approaches in terms of quick migration handover and less latency.

KEYWORDS: Cloud Computing; Edge server; Mobile-edge computing; Multimedia Streaming; Resource Allocation; Virtualization.

INTRODUCTION

In the past few decades, innovation has served mankind by giving manageable specialized answers for the social issues challenged by the society. The Internet of Things (IoT) will be realized by connecting billions of resource-limited mobile devices, sensors and wearable computing devices, to the Internet via cellular networks [1]. In recent years, the research groups have been working towards upgrading the innovative foundation and augmenting the proficiency of administrations for citizens to meet their changing requirements for more quick and smart living. Society has developed, and in the present period of smart phones, we have a new concept, called the smart environments for example “smart cities,". Smart environments are required to enhance the quality of life for their people, by utilizing the advancements of the Information and Communication Technologies (ICT). Smart environments are anticipated that would give the people with an assortment of imaginative services, ranging from education and medical services to expand and augmented immersive reality; for example, the tourism service can be very much improved by ICT. To be sure, conveyed benefits in smart environments will include not only smartphones and tablets, but also energy meters, washing machines, fridges, sensors for natural and environment observing and many more ; In short, the diverse parts of the Internet of Things (IoT) ecosystem. As illustrated in Figure 1, mobile-edge computing can provide the functionalities of cloud-computing at the edge of pervasive radio access networks in close region to mobile users.

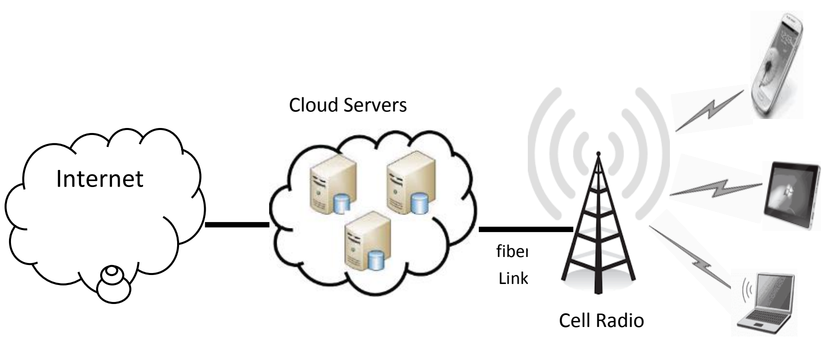

Figure 1. A typical scenario of Mobile Edge Computing

The target of the next generation smartphone systems, commercially known as the fifth generation (5G) is to speed up the development of smart cities, by not only increasing the data delivery rates but also deploying more numbers of IoT devices to be used by smart city services and applications [3][5][21]. Moreover 5G technologies will be able to support more numbers of smart services, which cannot be supported by current network architectures [6][9]. This includes real time and tactical applications, and services with high computing requirements such as of very-short latency and high responsiveness. 5G systems are depending on the novel technologies such as Network Function Virtualization (NFV), Software Defined Networking (SDN), and cloud computing to accomplish the system's flexibility and true elasticity [3][9]. Even though cloud computing has extremely advanced varieties of services; it is limited against emerging applications (e.g., Augmented reality) which require very quick response time. Cloud is also limited against the applications which require more computation power/CPU-constrained user equipment (e.g., mobile gaming) that need to partially run their computation in the cloud while ensuring response times in the range of milliseconds. These limitations are caused by the centralized architecture of cloud computing. However we could find a solution for these issues by means of Mobile Edge Computing (MEC), by achieving the computing and storage capabilities at the edge of the network, as near as possible to the end-user[2][10]. MEC provides a vital solution to these limitations. Really, it transforms the cloud hierarchy by pushing computing resources in the proximity of mobile users at the edge servers. The integration of MEC and 5G, certainly will improve the quality of life of citizens in smart cities. In this paper we show how MEC will enable emerging services for smart cities, focusing on an augmented reality, which is for the support of tourism in smart cities. The overall objective is to demonstrate how high quality of service (QoS) can be maintained in spite of the mobility of users through the MEC, more particularly through the concept of Track the Edge Approach (TEA) which is similar in spirit to the Follow Me Cloud concept[3][11]. TEA guarantees that the all the services are constantly follow the user and that the user is always serviced from the nearby edge.

LITERATURE REVIEW

In the recent years MEC is attracting the industries and the research community [13]. Several important standardization procedures have been initiated and the European Telecommunication Standard Institution (ETSI) formed a new Industry Specification Group (ISG) in 2014 and came up with various industry specifications [8]. In [14] the authors proposed a two-hop network in which edge architecture improves the data transfer rate and throughput for multimedia streaming data streaming compared to remote cloud. In [15] the authors presented a scheme which exploits network assisted adaptive streaming applications for multimedia content delivery inside MEC to boost up the Quality of Experience (QoE). The researchers proposed an architecture in [16] with distributed parallel edges to rise the QoE for content delivery. In the proposed model in [17] the authors makes use of edges as caches along with proxies to store media content. It also implements computation offloading to increase the lifetime of mobile devices. In [12], edges are acting as independent small-scale data centers on their own and are used for multimedia streaming data caching and streaming. In all these above research works, MEC is supposed to be a promising solution for handling multimedia streaming data services. Its limitations in terms of resource control and instrumentation have also been highlighted as important challenges. In smart city scenarios, the dynamic service migration is also required due to the users' mobility. Most research works on the latter consider the usual cloud environments [3] [11]. In [18], the authors proposed the migration of mobile edges using a Markov decision process approach to decide the optimal solutions for service placement.

PROPOSED METHOD

The key idea behind the MEC is to locate storage and computation resources at the network edge, in the closed proximity of users. Accordingly, all the computations can be pushed from far remote cloud servers to the edge devices. By distributing the data-processing to the edge devices and speeding up the data streams through various techniques (i.e., caching and compression), MEC decreases the network traffic towards the core network. In addition, it helps in cut down end-to-end latency, enabling the offload of important computation load from power-constrained user equipment to the edge. As discussed in the European Telecommunications Standards Institute (ETSI) MEC initiative [4], edge computing shall enable new computation-intensive services and shall yield promising business models. It also represents a fault resilient solution for its decentralized architecture [12].

PROBLEM DEFINITION

To the best our knowledge, in the mobile cloud computing environment support for the mobility and support for migration of service in terms of multimedia streaming data caching and content delivery have not been addressed yet. In the rest of this paper, we explain and illustrate an innovative deployment scenario on delivering the multimedia streaming data and how it can be enriched using MEC in spite of the user's mobility.

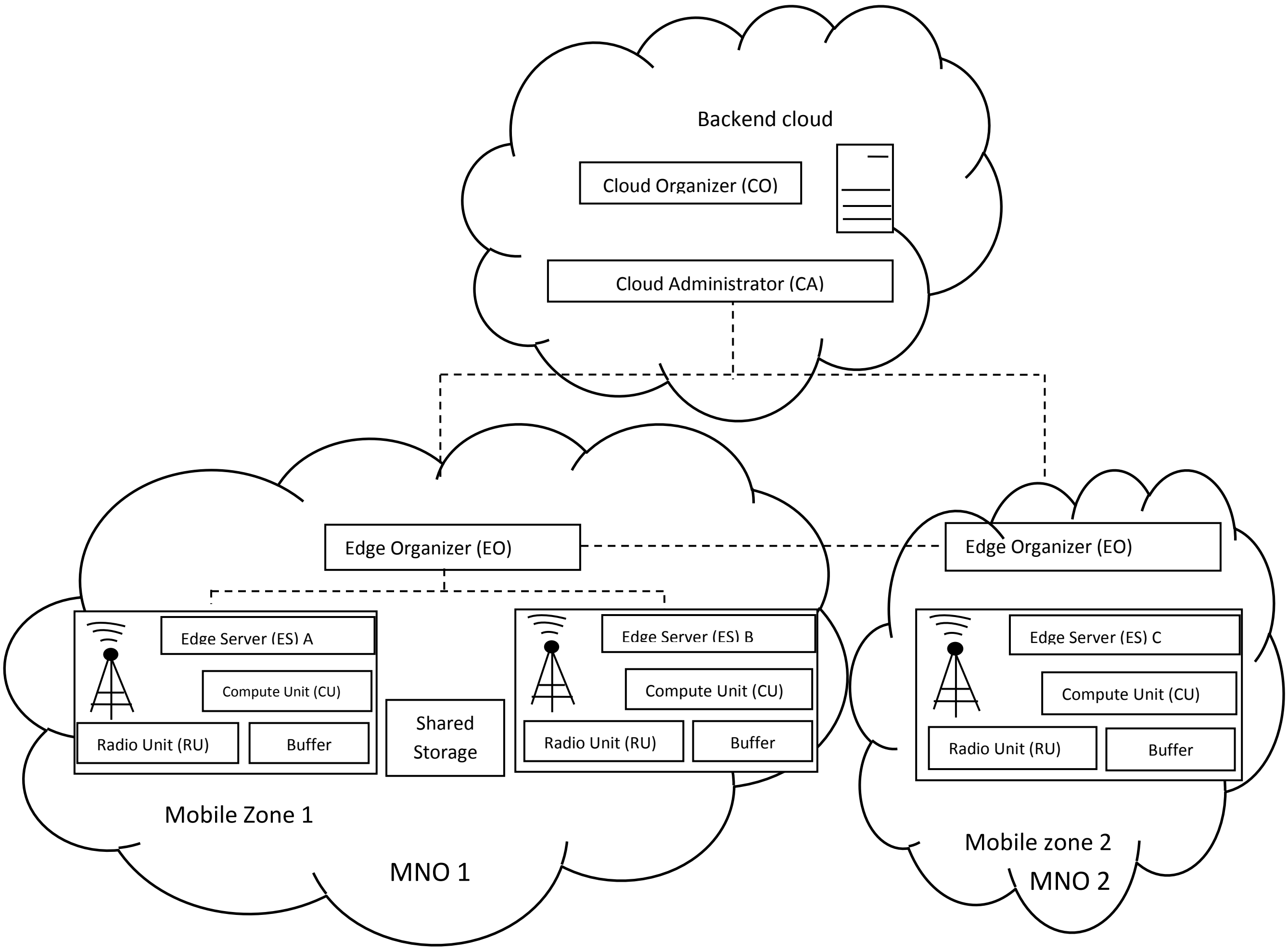

Figure 2. Mobile Edge Computing Architecture

The proposed architecture is based on the two-tier principle. In the first tier we have the cloud platform where as in the second tier we have the Mobile Network Operator (MNO). The MNO is functioning in the foot print of a cellular region called as Mobile Zones In this scheme the cloud service provider (CSP) provides access, using an Application Programming Interface (API) [19], to a content provider or a third party over-the-top (OTT) service to use the resources of the cloud to deploy its application. The cloud manages and integrates its own infrastructure and resources by using the cloud organizer. We have another component called the cloud Administrator (CA) which is used to maintain the service level agreement (SLA) with the OTT providers and MNOs. The edge server (ES) in the MNO's network is managed and controlled by Edge Organizer (EO). There is an EO for every MNO for managing its own set of ES clusters. The proposed MNO edge network architecture is depicted in the figure 2. The ES is located on virtual machines on top of the existing server hardware provided by the MNO's edge network node. The ES has its own Compute Unit (CU) and storage buffer facilities as separate nodes.

SYSTEM ARCHITECTURE

The CU node is responsible for providing container-based applications on the edge. The buffer is used to keep back-up of the application containers. In MEC container-based virtualization technique is considered as one of the most promising solutions [7]. Suppose a user wants to connect with a cloud to download or watch a multimedia streaming data he has to connect with the ES. Initially he will be served from the backend cloud server and the file will be downloaded with jitter and the resolution of the multimedia streaming data will be limited. In our proposed scheme, the EO initiates a container on the connected ES's Compute Unit with built-in streaming and transcoding virtualized functionality [19] [20]. Consequently, the subsequent contents of the multimedia streaming data file will be offloaded from the cloud server to the ES's local storage. After this, the user will receive the HD content of the multimedia streaming data stream from the ES. In the case of HTTP based streaming data streaming the EO can download the entire streaming data content or multiple chunks of the data in a single step at a time. After this the streaming data will be delivered to the user whose distance is one hop away from the ES, the quality of the streaming data is expected to be improved very much. For the OTT services the SLA which is implemented already is used to negotiate between OTT providers and the MNA for the quality of service to be provided to the user.

This setup is working well for providing the multimedia streaming data to the user as long as the user stays at a particular position. However, if the user starts moving and if the distance between the mobile user and the MNO increases, the user may start experiencing the degradation in the quality of the content delivered. To avoid this type of QoS issues and to maintain the same quality, it is important to make the content to move along with the physical mobility of the users in a method called the Track the Edge Approach (TEA). This TEA approach is similar to the Follow Me Cloud (FME) [11] approach used in cloud platform. To realize the TEA as a true real time system, the EO needs to maintain updated knowledge about all the resources and their positions. The position of the resources can be tracked using the MEC's active device location tracking system (ADLT) which provides the speed and direction of the moving device. Considering the data obtained by the ADLT system, the EO can calculate the latency between the user and the current edge and compare it with the latency between the same user and the destination edge. Once estimated properly, the EO can move the contents of the multimedia streaming data to the new ES. The user may then obtain the streaming data service from the new ES, which will confirm the reduction of latency and maintains the QoS. This process can be used whenever the user migration takes place. User migration can occur in different techniques. In the case of live multimedia streaming, continuity of service and minimum distraction are of major focus. To achieve seamless live migration, the state of the service has to be maintained in order to guarantee that no data is lost. This is realized by transporting the entire buffer content of the running instance (i.e., container) from the current ES to the destination ES. This is called as the handover process. The current ES keeps track of the modified memory blocks while the transfer is in progress.

Once this initial handover is finished, the modifications that have happened in the meantime are transferred again. This process is continued until the new instance becomes exactly similar to the old one. This guarantees that after the migration process is finished, the playing of media starts from the exact point without overlapping. However, data loss happens in the case of improper memory handling. This causes the overlap in the media playtime, so that the user has to watch the same content again. Furthermore, the duration of the migration should be within a time limit. If the duration is too long, it causes the user to be moved away from the ES or the media file would be almost over. To avoid these constraints, separate shared storage has to be maintained. If the memory blocks are stored at a shared location attached to the new ES, handover becomes faster rather than using the local storage.

EXPERIMENTAL SETUP

The prototype test bed model is shown in figure 3. The test bed is built to simulate the Mobile Edge-Based multimedia streaming and the user mobility. The test bed is built using one Ubuntu 14.04.3 Long Term Support (LTS) desktop system and two laptop computers with the same host operating system. We have used the Virtual box 5.0 to install Ubuntu on a desktop workstation machine to implement the test bed. The desktop machine hosts three virtual machines VM1, VM2 and VM3 inside the virtual box environment. VM1 is working as a gateway for the complete network to access the Internet. In VM2 the Devstack based cloud environment is implemented which provides all the required components like controller, compute, network, and storage on the same node with an Ubuntu instance running within it. The Ubuntu cloud instance uploads a HTTP Live Streaming (HLS) server. We have implemented the ffmpeg open source server as a separate VM, for both streaming and transcoding. The media contents (HLS fragments) generated from ffmpeg transcoding servers, and ffmpeg streaming server (hosted in a separate VM) is used to streaming the data. A floating IP address of the instance is selected from the same IP subnet range of the edge cluster, so the ES can easily access the data from the cloud VM. VM3 was configured using Proxmox, an open-source server virtualization environment (VE) and acts as edge cluster controller EO. A DHCP server is also included in VM3 with authentication. To automate the organization process, a script is used to:

- Monitor and control the session changes the clients from edge to edge using the authentication server logs.
- Control the migration of the container.

We have emulated the ESs by integrating two additional VMs (VM4 & VM5) which are implemented in two different laptop systems. An Ethernet switch is used to connect the network. Both VM4 and VM5 use the same VE as EO. The wireless LAN interface was configured using Host-apd (Host access point daemon) in IEEE 802.11 master mode in the laptops (VM4 and VM5) which are acting as edge access points. The container is created inside the VM4 and the Openvz- an operating system-level virtualization technology for Linux is used as containers for the test bed. We have used the Nginx as the web server and it is configured to serve as reverse proxy to the back-end cloud HLS server. Nginx supports caching and streaming functionality for the multimedia content. Our objective of these tests is to validate the use of MEC to ensure high-quality HD multimedia streaming service for the mobile users.

Therefore, the aim of these tests is on caching the multimedia content and the on time delivery of the multimedia content closer to the user at the edge to ensure the continuous play of the multimedia content at the user's mobile device. To conduct the experiment, an instance of one container is created in Edge1 with all the above said features. When a user connects to the network by using a mobile device like smartphone or laptop, the device uses the ***s**ervice **s**et **id**entifier (*SSID), to connect with the network. Then the user is assigned an IP within the same IP subnet pool of the ES. Immediately a log is created and saved with the MACID of the user in a database in the EO. The user opens a browser and starts watching the multimedia data, using the URL of the streaming sever which is hosted at the container. To implement the basic security, only authorized user is given rights to open the data from the container. When connecting for the first time, the request is forwarded to the cloud VM, and the multimedia content is delivered from the back-end cloud. The container starts caching the requited media contents and stores them for further use. If the same multimedia file is requested next time, the container itself serves it through its own streaming functionality by using cached contents irrespective of the accessibility of the cloud. This decreases the traffic in the core network, and the user is also served as the network and the cloud is always available freely. We have used laptops to test the user migration. Initially one laptop is placed near the first edge ES A and connected to VM4. The multimedia content is downloaded from the backend cloud (VM2) and it is played in the laptop. Then the laptop is gradually moved towards the second edge ES B. During this movement, the user is automatically connected to VM 5 and all the logs are handed over to the new edge server. This is done by collecting the user's MAC ID and compared with the database, to ensure that the same user is migrating from the first ES to the second ES. In Openvz, the live content delivery is carried on by checkpoint / restore in user space (CRIU). It executes the utility vz-dump which is used for memory block dump to retain the state information and utilizes the incremental file transfer utility rsync to move the content to the migrated location. The duration of migration of one container is shown in figure 4. We have taken three different situations for the experimental purpose:

a) Streaming in use - online mode — User is watching the multimedia file with active streaming

b) Streaming is not used - offline mode — User is not watching the video but streaming is active

c) Empty container with two different types of ES

Further in every situation the ES is used with different memory (RAM) configurations. The experiments were conducted with ES containing 1 GB, 2GB and 4 GB RAM.

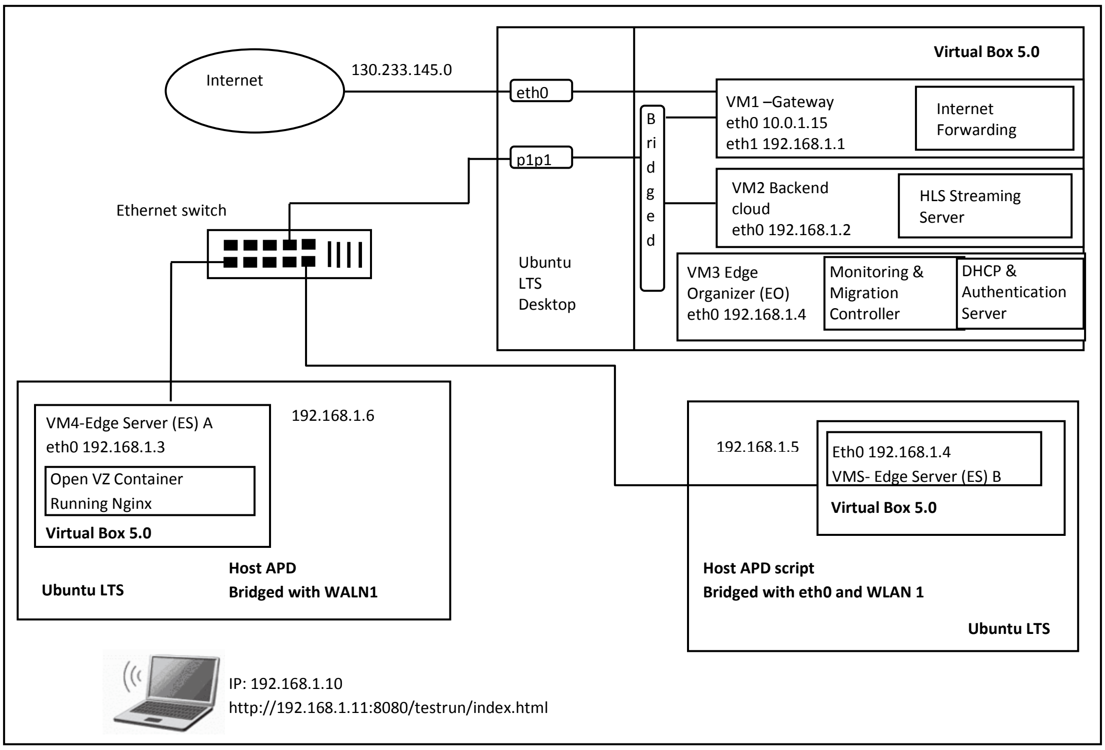

Figure 3. Mobile Edge Computing test bed setup

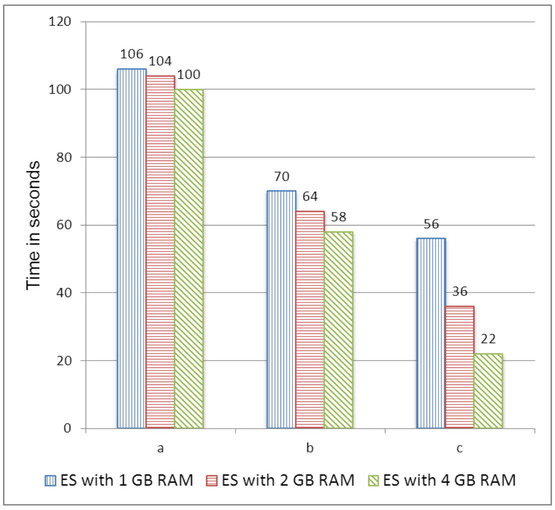

Figure 4. Duration of migration with local storage (a) Active streaming (b)Inactive Streaming and (c) Empty container

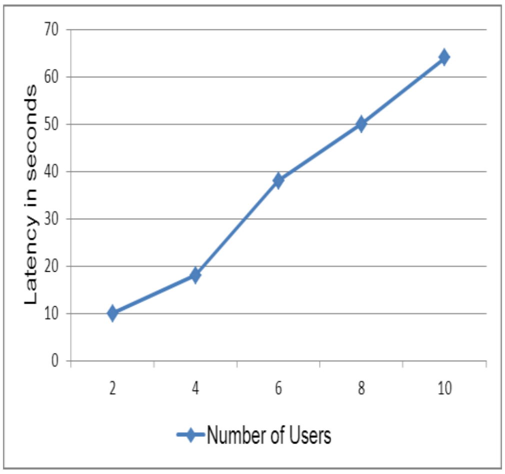

Figure 5. Latency of migration with different number of users

Figure 5 shows the latency in delivering the streaming media when more number of users added with the same ES. The latency slowly increases when number of users' increases. But this depends on the size of the container. If we configure the ES with big container, the performance can be improved further. The results of these experiments illustrates that the type of the storage and memory capacity highly affect the migration latency.

CONCLUSION AND FUTURE RESEARCH DIRECTIONS

In this paper, we proposed a novel user orient mobility management scheme to enhance QoS of Mobile Edge Computing. We have developed the TEA approach, which enables a very quick handshaking process for mobile devices to support ultra-quick real time applications. Our approach will play a vital role in developing smart environments like smart cities. The Proposed framework is tested using a real time test bed under various configurations like different storage sizes, different container sizes and different numbers of mobile users. The results demonstrate that the latency of migration depends on the various real time conditions. We have obtained interesting results, which suggests migration latency depends on the different techniques used. Based on the results obtained, it can be concluded that a strategy has to be used to select the right mixture of techniques to be used for efficiently migrating important applications. This will be considered as one of the authors' future research directions in this area.

REFERENCES

[1] M. Swan, "Sensor mania! The Internet of Things, wearable computing, objective metrics, and the quantified self 2.0," J. Sens. Actuator Netw.,vol. 1, pp. 217–253, 2012.

[2] "ETSI GS MEC 003 Mobile Edge Computing: framework and reference architecture v1.1.1," *ETSI MEC ISG, Tech. Rep.,* 2016.

[3] T. Taleb, A. Ksentini, and A. Kobbane, "Lightweight Mobile Core Networks for Machine Type Communications," IEEE Access, vol. 2, Oct. 2014. pp. 1128–37.

[4] https://portal.etsi.org/portals/0/tbpages/mec/docs/ mec_executive_brief_v1_28- 09-14.pdf

[5] T. Taleb and A. Kunz, "Machine Type Communications in 3GPP Networks: Potential, Challenges, and Solutions," IEEE Commun. Mag., vol. 50, no. 3, Mar. 2012.

[6] T. Taleb et al., "EASE: EPC as a Service to Ease Mobile Core Network," IEEE Network, vol. 29, no. 2, Mar. 2015, pp. 78–88.

[7] C. Pahl and B. Lee, "Containers and Clusters for Edge Cloud Architectures--A Technology Review," in *IEEE International Conference on Future Internet of Things and Cloud (FiCloud)*, 2015.

[8] https://portal.etsi.org/portals/0/tbpages/mec/docs/mobile-edge_computing_introductory_technical_ white_paper_v1_18-09-14. Pdf

[9] T. Taleb, "Toward Carrier Cloud: Potential, Challenges, and Solutions," IEEE Wireless Commun., vol. 21, no. 3, June 2014, pp. 80–91.

[10] U. Shaukat et al., "Cloudlet Deployment in Local Wireless Area Networks, Motivation, Taxonomies, and Open Research Challenges," J. Network Computer Applications, vol. 62, Feb. 2016, pp. 18–40.

[11] Ksentini, T. Taleb, and F. Messaoudi, “A LISP-Based Implementation of Follow Me Cloud,” IEEE Access, vol. 2, Oct. 2014. pp. 1340–47.

[12] H. Chang et al., “Bringing the Cloud to the Edge,” Proc. IEEE INFOCOM Wksps.), Toronto, Ontario, Canada, May 2014.

[13] Ahmed and E. Ahmed, “A Survey on Mobile Edge Computing,” Proc. IEEE 10th Int’l. Conf. Intelligent Systems Control, India, May 2016.

[14] D. Fesehaye et al., “Impact of Cloudlets on Interactive Mobile Cloud Applications,” Proc. IEEE 16th Int’l Conf. Enterprise Distributed Object Computing, Beijing, China, Sept. 2012.

[15] J. Fajardo, I. Taboada, and F. Liberal, “Improving Content Delivery Efficiency through Multi-Layer Mobile Edge Adaptation,” IEEE Network, vol. 29, no. 6, Dec. 2015, pp. 40–46.

[16] W. Zhu et al., “Multimedia Cloud Computing,” IEEE Signal Processing Mag., vol. 28 , no. 3, May 2011, pp. 59–69.

[17] Y. Jararweh et al., “Resource Efficient Mobile Computing Using Cloudlet Infrastructure,” Proc. IEEE 9th Int’l. Conf. Mobile Ad Hoc Sensor Networks, Dalian, China, Dec. 2013.

[18] S. Wang et al., “Dynamic Service Migration in Mobile Edge- Clouds,” Proc. IFIP Networking Conf., Toulouse, France, May 2015. [m]

[19] P. Frangoudis et al., “An Architecture for On-Demand Service Deployment over a Telco CDN,” IEEE ICC ’16, Kuala Lumpur, Malaysia, May 2016.

[20] T. Taleb, A. Ksentini, and R. Jantti, “Anything as a Service for 5G Mobile Systems,” to appear, IEEE Network.

[21] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, “Five disruptive technology directions for 5g,” IEEE Communications Magazine, vol. 52, no. 2, pp. 74–80, 2014.