Noname manuscript No.
(will be inserted by the editor)

Multi-Access Edge Computing: Open issues, Challenges and Future Perspective

Sonia Shahzadi · Muddesar Iqbal · Tasos Dagiuklas · Zia Ul Qayyum

Received: date / Accepted: date

Abstract Latency minimization is a pivotal aspect in provision of real time services while adhering to Quality of Experience (QoE) parameters for assuring spectral efficiency. Edge Cloud Computing, being a potential research dimension in the realm of 5G networks, targets to enhance the network efficiency by harnessing effectiveness of both cloud computing and mobile devices in user's proximity. Keeping in view the far ranging impact of Edge Cloud Computing in future mobile generations, a comprehensive review of the prevalent Edge Cloud Computing frameworks and approaches is presented with a detailed comparison of its classifications through various QoS metrics (pertinent to network performance and overheads associated with deployment/migration). Considering the knowledge accumulated, procedures analysed and theories discussed, the paper provides a comprehensive overview on sate-of-the-art and future research directions for multi-access mobile edge computing.

Keywords Cloud Computing · Edge Cloud · Mobile Cloud Computing · Internet of Things · Cloudlets · Fog Computing

1 Introduction

The forthcoming emergence of Internet over Everything is driven by the evolution of 5G communication, rapid growth of cloud, social media, and mobile computing, the use of Data Science to generate smart analytics value. This evolution brings to the forefront new type of communications such as Machine to Machine and Person to Machine [3]. According to RCRwireless, more than 50 Billion IoT devices will be interconnected by 2020 [6]. In this new environment, there is a need to manage, process and store the huge amount of data generated at the network edges. Cloud computing frees the enterprise and the end users from many details. As an effect, computational and network overhead at central cloud increases. This creates problems with real-time applications where latency is a crucial factor. Edge Cloud (EdgeC) Computing has been introduced to reduce network stress (i.e. latency) by shifting resources at the edge of network to proximity of mobile users and IoT while providing services and seamlessly processing the contents. As it implies, the idea of EdgeC has emanated from Cloud Computing (CC) leading towards to Mobile Cloud Computing (MCC). It offers cloud resources at the edge of network with low latency and high bandwidth. MEC started to gain attention of research community in last few years with preliminary research contributions so far such as: standardization of some key interfaces for mobile edge computing [2], building super-short applications requiring a low response time and latency [62] [7] [4] and modeling hyper-scale datacentres with micro datacentres at the edge of networks [9]. Executing the computing-intensive applications consume lots of power at the mobile device. The advances in EdgeC have made it possible to provide infrastructure, platform, and software as a service for the

Sonia Shahzadi
Swan Mesh Networks Ltd, UK.
E-mail: s.shahzadi@swanmesh.com

Muddesar Iqbal
London South Bank University, UK.
E-mail: m.iqbal@lsbu.ac.uk

Tasos Dagiuklas
London South Bank University, UK.
E-mail: tdagiuklas@lsbu.ac.uk

Zia Ul Qayyum
University of Gujrat, Pakistan.
E-mail: vc@uog.edu.pk

end-users from any computer with a fixed or wireless Internet connection. EdgeC can extend such services to mobile devices. Since there are several billions of mobile subscribers world-wide, EdgeC has the potential to have far-reaching impacts in the wireless industry and in our society. The delivery of demanding applications (e.g. streaming, augmented reality, on-line gaming etc) to/from the cloud to the mobile users relies on wireless networks (e.g., WiFi, 3G, 4G, 5G etc) for data and control between the cloud and mobile devices. Compared with fixed networks, wireless networks have limited bandwidth, latency due to network congestion, and connectivity. Moreover, under the presence of more mobile devices, the bandwidth available to each device will be further reduced, and network latency can go up and response time for mobile users can be higher. The main objectives of this paper are to present use case scenarios associated with Edge Cloud Computing, describe the latest advances in different standardization fora related to Edge Cloud Computing, discuss future research challenges. In remainder of this paper, section 2 describes the application scenarios and motivation; section 3 describes different approaches for Edge Cloud computing and section 4 describes the classification and comparison of existing frameworks. The section 5 presents the open research challenges and issues and paper is finally concluded in section 6.

2 Motivation

As an effect, the initial objective of EdgeC is to adapt cloud computing to the mobile environment in anywhere and anytime manner, where data are stored and processed outside mobile devices [74] [55]. Some of the most critical issues related to EdgeC include: network latency and limited network bandwidth and user mobility. Despite the advances in smartphones, they still have limited processing capability and limited battery life, especially with the growing demand for energy-hungry applications, such as video streaming and 3D gaming. [72] described MEC as an emerging paradigm that provides computing, storage and networking resources within the edge of mobile Radio Access Network (RAN). The preparation for deployment of 5G network and tactile Internet sparked conversations about issues that need to be solved to increase the QoE of applications based on this platform. These applications require low latency and real-time data to effectively utilize its functionalities. Research done by [78] observed that the existing cloud infrastructure cannot resolve this issue. [15] explained the key issue mobile edge computing aims to solve is to reduce the network bandwidth and latency in other to improve QoE. This would be done by bringing cloud infrastructure closer to the user. [27] demonstrated that deploying cloudlets in close proximity with the end user improves the execution of latency-critical applications.

The trend of pushing cloud computing to the edge of mobile networks are expected to continue to accelerate in years to come. According to [72] the challenges and open issues associated with MEC includes data interoperability, resource management, and orchestration, service discovery and security.

This section present scenarios are highlighted in the following section where MEC can be beneficial in terms of performance improvements [7].

2.1 Augmented Reality (AR)

Augmented reality (AR) merges the view of real world and computer generated sensory inputs such as graphics, GPS data, sound and video [7]. AR allows the user to see the real world, with virtual objects superimposed upon or composited with the real world so that the information about the surrounding real world of the user becomes interactive and digitally manipulable. EdgeC can be used for generation of rendering. Required processing can be performed on EdgeC instead of the main server due to requirement of high processing speed and low latency.

2.2 Connected Vehicles

The number of connected vehicles has been increased to support Vehicle-to-everything (V2X) communication, e.g. inform vehicles about road conditions through image/video analysis route prediction, collision warning applications such as safety, infotainment and communication or any other information that may affect the vehicle. Furthermore, the use of Roadside Units is intended to increase efficiency, and convenience of the V2X applications [85]. As the number of connected vehicles increases and use cases evolve, the volume of data will continue to increase along with the need to minimize latency and optimize QoE. EdgeC can be very useful to push V2X applications, data, and services from central cloud to the edge of network (e.g. Roadside units), this would help in bringing data and analytics applications closer to the vehicles at the roadside units, enabling applications acceleration over the vehicles [81]. The Mobile Edge Computing application can operate as a highly distributed roadside unit to support vehicle-to-everything (V2X) communication. Thus helps in sending the useful information to the nearby cars without any delay. This Instant communication can help drivers to react in timely fashion in order to avoid accidents and improve road safety.

2.3 Internet of Things (IoT)

IoT is a network that connects physical devices, sensors, vehicles and everyday electronic objects embedded with software, actuators and sensors to collect and exchange data, but also goes beyond this to include connections and networking between transport services, community services and much more of societys infrastructure [40]. The IoT is the latest technology and it is as important as the Internet. It is a network that connects all things to the Internet for exchanging information and communication through devices with agreed protocols by identifying, locating, monitoring and managing things [22]. In other words, the Internet is no longer bound by the desktop, but goes out into the world of other things [40]. The enormous amounts of data generated by this process would be best stored on a cloud. Moving IoT application data to the cloud can reduce the cost and complexity that relates to hardware management [20]. There is a need to aggregate various IoT device messages using mobile cloud computing closer to the device users to improve latency and response time. Various devices are connected over different forms of connectivity, such as 3G, 4G, 5G, Wi-Fi or other radio technologies [28]. In general, the messages are small, encrypted and come in different forms of protocols (e.g. MQTT, CoAP etc). There is a need for a low latency aggregation point to manage the various protocols, distribution of messages and for the processing of analytics from data collected from different IoT deployments [30]. The EdgeC server provides the capability to resolve these challenges.

2.4 Edge Cloud Media Optimization

The distributed Edge Clouds have been designed and developed to support the media services across heterogeneous wireless and converged networks. The EdgeC provides support to the immersive applications to handle challenges such as user mobility and scarce network resources. It also helps in developing cloud-based workflow management for media applications, intelligently serving the end users through the available communication capacity and end-user device capabilities. This necessitates carrying media related functionalities such as rate adaptation/transcoding, rendering and caching as shown in figure 1.

This use case aims to optimize QoE for video applications over radio access network. This can be accomplished by estimating throughput at the radio downlink interface from radio analytics information. EdgeC can be used to enhance QoE for the users by to adopting a video using application-level coding (e.g. transcoding, rate control) to matches the estimated capacity at the radio downlink [23].

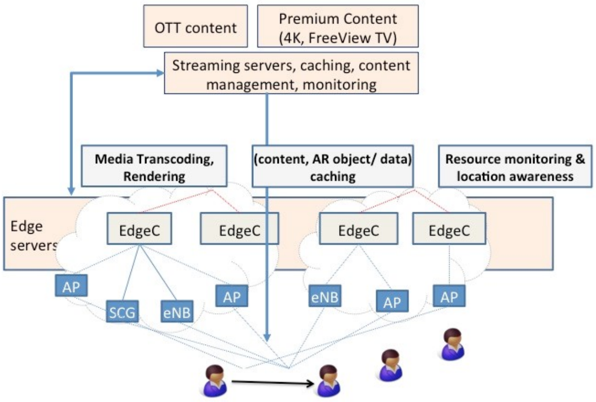

Fig. 1 Edge Cloud Media Optimization

The adoption of Edge Cloud could be adopted by network operators either at the access networks (e.g. Evolved Node B (eNB), Wireless APs) or at the aggregation points that are interconnected to the core networks via backhauling. A major challenge is that the user mobility may affect the entire operation when he/she is associated in an AP where the network operator has not deployed. In this case, the operator may seek for on-the-fly computing resources by requesting the available computing resources in an on-demand fashion from third parties. Such third parties could be any foreign cloud provider with enough/available computing resources that could be rented on demand by the network operator. This necessitates the establishment of an agreement between the involved parties through a federation scheme. An immense volume of resources possibly with different specifications need to be managed under a unified and federated framework in terms of physical nodes and their accompanying physical and virtual resources. Once the requested computing resources are transferred to the network operator, another task is to manage computing resources. Due to mobility, the user may be associated (using handover management) to an area where neither the AP nor the aggregated node has EdgeC resources. In this case, the computing resources that will be requested by the third party can be established. After this establishment, cloud computing resources must be transferred from the old EdgeC to the new one. This scenario is illustrated in the figure 2.

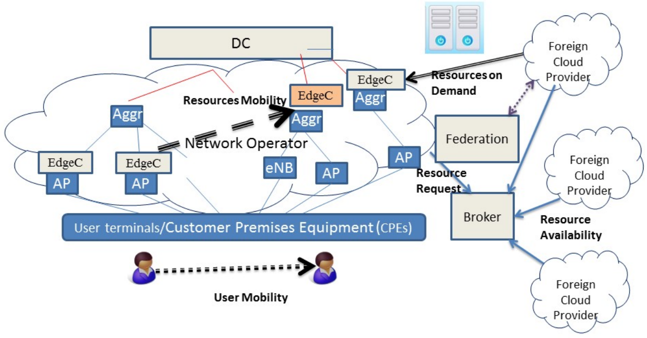

Fig. 2 Distributed Cloud Federation and Service Brokerage Model

3 Edge Cloud Computing Standardization and Fora

In this section, some relevant approaches presenting similar concept are comprehensively elaborated. These approaches are broadly categorized into (1) MEC based approaches (2) Cloudlets based approaches and (3) Open Fog Consortium.

3.1 MEC approach

ETSI introduced Mobile Edge Computing (MEC) in ETSI [2] which is designed to push resources closer to the radio access networks in 4G and 5G. It brings cloud-computing capabilities and IT service environment at the edge of the mobile network. This environment is characterized by ultra-low latency and high bandwidth as well as real-time access to radio network information that can be leveraged by applications. Since 2017, the ETSI MEC industry group has renamed "Mobile Edge Computing" to "Multi-Access Edge Computing" to better reflect the growing interest in MEC from non-cellular operators. MEC has the aim to reduce network stress by moving resources from cloud to mobile edge [17] [16]. Fully virtualized MEC infrastructure is proposed in [16]. A distributed computation offloading algorithm is presented in [24]. SEcS (Scalable Edge computing Services) framework is presented to build and deploy Edge computing Services [39] to address the challenge of scalability, high availability, fault tolerance and robustness [39] Multi-access MEC architecture is designed to addresses latency and bandwidth issues for the video analytics location services, Internet-of-Things (IoT), augmented reality, optimized local content distribution and data caching and many other use cases and application scenarios for Smart Cities, Healthcare, Disaster Management and Smart farming. A scheme is proposed in [32] to deal with unpredictability of computation availability at the edge, where task execution performed on idle edge resources. In [31], it argues that autonomic computing techniques are fundamental element for dynamic management of edge servers. MEC architecture is proposed in [84] to reduce latency. To migrate running application from VM or container for mobile edge cloud environment, a layered solution is proposed in [59]. A WiCloud is proposed in [54] that is based on NFV/SDN concepts to provide edge networking. Some frameworks are given in [14] for mobile application execution in MCC and their comparative study are also given. Seamless application execution frameworks in MCC are also highlighted in [13] with detailed comparisons and analysis.

3.2 Cloudlets based approach

The term cloudlet was coined by researcher at Carnegie Mellon University, where its prototype is developed as part of a research project [1]. The Cloudlets are designed to support applications for mobile devices those are resource hungry and interactive e.g. Augmented reality applications, Cloud games, Wearable cognitive assistance system Google Glass, Apple Siri and Google Now and many other applications those require offloading of resource intensive task from cloud to the mobile device to achieve the required quality of experience. This helps in reducing communication latency and perform faster execution for application intends to perform resource intensive tasks. The main motivation of cloudlet comes from the Internet community to handle the resource constraint on the mobile devices.
A cloudlet represents the middle tier of a 3-tier hierarchy i.e. Mobile device, Cloudlet, Cloud. Cloudlets can be considered as a local data centre in a box to enable localized cloud services, offer high performance and faster access to cloud resources by multiple users simultaneously [46]. Moreover, it deals with large WAN latency, less bandwidth, and high utilization cost issues

[46].

Cloudlet, through the interest of key industrial players (e.g. Nokia, Intel, Vodafone) have formed the open source banner of Open Edge Computing (OEC) Initiative. OEC has offered cloudlets open-source code APIs as an extension to OpenStack to promote cloudlet as an enabling technology [5]. The main goal is to, engage with wider IT and Telecoms industry to Synchronize the work with other efforts includes ETSI ISG MEC and OPNFV. The cloudlet pioneering Elijah project at Carnegie Mellon University has been extended to OpenStack++: to provide a cloudlet library based on a modified QEMU with integration into the OpenStack platform.

A mesh cloud architecture is proposed in [46], which is composed of cloudlet, Internet cloud and wireless mesh networks. An experimental framework is designed in [47], in which private cloudlet and wireless mesh network is implemented. It is capable of establishing and maintaining mesh connectivity among multiple nodes automatically and is featured with adaptivity and self-recovery in case of network failures [47].

Cloudlet architecture presented in [76], manages applications at the component level. Cloudlet based MCC system is introduced in [44] for reduction of power consumption and network delay. A Performance Enhancement Framework for Cloudlet (PEFC) is proposed for MCC [79] to improve the cloudlet performance. Centralized cloudlet architecture is proposed in [66].

In [76], a new cloudlet architecture is proposed where applications are dealt on a component level where components are distributed among dynamic cloudlets letting users to join and leave cloudlets at runtime. However, cloudlet performance relies upon user mobility [56].

In [21], two migration models have been compared cloudlet network design i.e. VM bulk migration and VM live migration.

In order to access discoverable cloudlet server for mobile users for resource provision and services on demand, a cloudlet system is proposed in [65]. These cloudlets may be deployed at various public places where users can connect cloudlet through a mobile network provider [65].

Another cloudlet based system is proposed in [79], with focus on performance improvement in mobile cloud computing. Cloudlet is installed along with AP to allow mobile devices to access it. These mobile devices connect to nearby cloudlet using Wi-Fi [44].

To minimize, delay and power costs of mobile user, cloudlet infrastructure is proposed in [43]. In [61], an energy management approach is introduced for mobile /pocket cloudlet. Researchers and Marine Corps are working together on the concept of tactical cloudlet. to implement distributed cloud computing concept in a remote and mobile battlefield scenario, especially in a more hostile environments e.g. during a war mission or disaster recovery where the requirements for communication changes quickly and requires higher power for computing [8]. Tactical cloudlets are proposed in [53] to support tactical edge and cyber-foraging where resource intensive tasks are offloaded to cloudlets. A strategy is proposed in [57], to reduce multi-resource allocation problem between cloudlet and mobile devices that will enhance Quality of Service (QoS).

3.3 Open Fog Consortium

Fog Computing is a concept introduced by Cisco in 2011 to meet the demands from different segments of Internet of Things (IoT), Internet of Everything (IoE) or Internet of Me (IoM) start to take off, e.g., consumer, wearable, industrial, enterprise, automobile, healthcare, building, energy. Classical cloud computing paradigm can hardly satisfy low latency, mobility support and location awareness. To address these problem, fog computing paradigm is introduced which improves quality of services (QoS) for real time applications and streaming, provides low latency and location awareness [70] in the field of wireless sensor networks, industrial automation and transportation systems. The main motivation is to alleviate the disadvantages of cloud computing: Long WAN latencies is a big obstacle in the critical path of user interaction and can deteriorate usability, traffic to central cloud increases computational and network overhead at central cloud. Fog computing introduces decentralized computing infrastructures so that computing resources and applications services are distributed in the most logical, efficient places, at any point along the continuum from the data source to the cloud. The main emphasis of the open fog consortium is to define a system-level horizontal architecture that distributes resources and services of computing, storage, control and networking anywhere along the continuum from Cloud to Things [19]. It's put data close to the end user [68] which reduce latency and improve QoS [71]. It also enables localization, context awareness and mobility support [70]. The Decoy Information techniques are used to detect malicious attacks those cannot be addressed using traditional security measures such an attack by an insider by seeding data into a system which appears genuine but actually it is fact spurious. Using Decoy information technology, you can implement security in fog computing [71]. Fog provide high quality streaming through access points and proxies to mobile nodes including moving vehicles [10]. It is suitable for those applications that require predictable and low latency such as video conferencing and gaming [18]. Fog architecture is given in [58] and radio access network (F-RAN) based fog is presented in [63]. In fog computing, cloud resources such as compute and storage etc. are migrating to the edge of the network where routers themselves may become the virtualized infrastructure [74] and services can be hosted at end devices such as set-top-boxes or access points [70]. In addition, multiple heterogeneous decentralized and ubiquitous devices communicate and cooperate with each other and can perform processing and storage tasks using network without the interference of third-parties [74]. It provides highly virtualized platform that offers storage, computing and networking services between the main cloud data centers and end devices [55] . It supports multiple services and applications where low latency is required. Fog /edge nodes have sufficient computing power to facilitate users task that are received from their end devices. This edge computing concept is introduced within cloud to reduce end-to-end response time between multiple devices. Although cloud computing provides lot of benefits to users in terms of cost reduction, system administrative tasks eliminations, flexibility increase, and improve reliability etc. but it also suffers some limitations including unpredictable network latencies and security issues etc. To overcome these limitations, fog computing is introduced where cloud system is located at the edge of network [35]. Fog computing will be helpful for emerging network paradigm which requires faster processing with less delay [35]. It is able to provide high quality streaming to mobile users through access points or proxies [10]. It is suitable for video streaming, gaming and augmented reality where low latency is required [10]. Fog computing not only reduces latency but also improves the QoS [68]. In fog computing paradigm, data is distributed and moved closer to the end-user and also support for data streaming and mobile computing [71]. Fog is considered to address services and applications that not well fit in cloud, e.g. video conferencing and gaming applications that need predictable and low latency, fast mobile applications, smart grid and smart traffic light system etc. [18]. In short, the aim of fog computing is to place cloud resources, close to mobile users [58]. A FSDN is proposed in [73], which combines the Software Defined Networking (SDN) and Fog computing.

ETSI-MEC consortium is developed to unite the IT Cloud and Telecommunication industry on MEC standards to providing IT and cloud-computing capabilities within the RAN through Mobile Orchestrator APIs for provisioning and monitoring virtual resources, targeting especially network function visualizations. Considering the overlapping interests in MEC and Cloudlets, a few on the other hand, Cloudlet/OEC have been motivated by the Internet community to optimize Internet demanding applications over resource-constrained mobile devices. Fog computing is mainly driven by IoT and the need for data processing and interoperability at the edge. All these three approaches presented have been compared in terms of different quality parameters in [5] are presented in table 1.

4 Comparative Analysis of Existing Frameworks

We have compared existing frameworks on the basis of various properties. Caching is used to store data locally to reduce the delay [13]. In case of cloudlet, it improves the latency by minimizing delay. Mirroring is also used to cache data at the mirror during uploading and downloading which also reduce the delay [83]. It reduces the operational overhead and optimizes response time but it increases the cloud storage cost [13]. Parallel execution improves the execution time of an application but it increases power consumption and hardware cost [13]. Pre-installations improves runtime data transmission but it increases the cloud resource consumption and maintenance overhead [13]. The optimize VM migration enables migration for only relevant applications instead of whole VM migration which reduces transfer overhead and improves the transmission time [13]. Fault tolerance provides a transparent mechanism to failure detection which requires a continuous monitoring but can create high complexity [13]. The reduction in the number of hop distance results in improved latency, jitter and response time.

CloneCloud, a flexible application partitioner is proposed in [25]. It automatically distributes the computation tasks from single mobile device to multiple machines. It significantly improves task processing and reduce the energy of smart mobile devices. Although it provides optimal execution time for computation environment but it increases the data transfer overhead on multiple machines. A framework is proposed in [80], to perform the optimal dynamic partitioning and execution of applications. It provides high performance with low operational cost but it also increases the data transfer overhead. Another solution is proposed in [77], for dynamic adaptive deployment of applications to enhance the quality of service. It offers optimal deployment of applications but it depends on nearby servers. Hyrax, a platform to support mobile devices is derived from MapReduce [60] and provide infrastructure for mobile computing. It improves utilization of mobile resource but execution of MapReduce jobs on mobile phones

Table 1 Comparison of Cloudlets, Fog and MEC approaches [5]

Properties	**Cloudlets based approach**	**Open Fog Computing**	**MEC approach**
Reduce Latency	*Y*	*Y*	*Y*
Reduce Jitter	*Y*	*Y*	*Y*
Multi-Tenancy	*Y*	*Y*	*Y*
With virtual IaaS platform?	*Y*	*Y*	*Y*
Co-Location	Y	Y	Y
Geographical Distributed	*Y*	*Y*	*Y*
Mobility Support	*Y*	*Y*	*Y*
Inspired from	Tactile Internet	IoT	Mobile World
Extended from Cloud	*Y*	*Y*	May or may not
Mostly used with wireless access	May or may not	*Y*	*Y*
Focus on-line analytics	May or may not	N	Y
Located between DC and device	*Y* but can directly run on a device	*Y*	*Y*
Improve User Experience	*Y*	*Y*	*Y*
N-tier	N=3	N=3 or more	N=2 or 3

Y=Yes; N=No;

results in high overhead for devices with limited resources. The work presented in [26] offers infrastructure deployment through CloneCloud architecture for smartphones applications to boost the mobile applications via cloning by using multiple computing platforms. It overcomes the limitations of mobile resources and the clone can be used as a recovery processing but it increases computation transformation overhead. The work presented in [29] provides highly dynamic and energy saving solution and is optimal for latency sensitive applications but has high profiling overhead. VM based cloudlets with one-hop access to improve the response time of applications is presented in [67]. Two types of algorithms, ALL and K-step are proposed in [36] to improve the static and dynamic partitioning of cloud applications. It provides an optimal and transparent solution for distribution of different application modules and significantly improves the performance of cloud applications but it is not highly flexible yet. A framework is proposed in [42], to execute the mobile applications on the cloud virtualization environment where the user can control the deployment and execution of the application. Cloudlet Aided Cooperative Terminals Service Environment (CACTSE) is proposed in [64] for mobile content delivery service where Mobile terminals are connected with each other via Service Manager (SM) which acts like a cloudlet module to improve the user experience. Cloudlet based dictionary for mobile devices is proposed in [11] with support for translation of 6 languages. It is easily configurable and extensible but requires high processing power for fast computation. The work presented in [76], offers a dynamic cloudlets concepts. A virtual mobile cloud computing provider proposed in [41] is a resource friendly architecture. To reduce the computational workload on smartphones, mirroring approach is proposed in [83] that takes a mirror against each smartphone and virtually expand smartphones resources. COMET (Code Offload by Migrating Execution Transparently), a runtime offloading environment towards augmenting smartphones is proposed in [37] to improve computation speed but it consumes more bandwidth. A framework is proposed in [52], to support seamless mobile cloud applications execution that significantly reduces latency and power consumption. Cuckoo, a dynamic runtime system for computation offloading [45] is suitable for compute intensive operations. MOCHA (Mobile Cloud Hybrid Architecture) with mobile-cloudlet-cloud architecture is proposed in [69] for real time face recognition that gives the minimum response time. AIOLOS, a mobile middleware is proposed in [75] which improves the mobile application performance via cyber foraging and optimize execution time and energy consumption. To enable the seamless and transparent usage of cloud resources, an elastic application platform is proposed in [82] that will augment the computing capabilities of mobile devices and provides elasticity between cloud and resource constrained devices. ThinkAir is an on-demand resource allocation framework with dynamic scaling [48] where users can migrate mobile applications to the cloud and it optimizes execution time and energy consumption. Pocket Cloudlets is proposed in [49] that replicates the search and advertisement based on personalized user behaviour and improve mobile user experience. Misco, a MapReduce framework is proposed in [33] for mobile devices. It supports any device with network connectivity and support for python. XMPP-based architecture is proposed for dynamic partitioning of mobile applications deployment between cloud and mobile devices and it offers flexible and extensible architecture [50]. Mobile Augmentation Cloud Services (MACS) middleware is presented in [51] which enables adaptive application partitioning of Android services and computation offloading. It reduces local execution time. A lightweight secure cyber foraging is implemented in [38] which are useful for resource, constrained devices. It enables new applications without a new hardware investment. Cloudlet based network is proposed in [34], It considers the impact of cloudlet in interactive mobile cloud computing applications and reduces data transfer delay. Later, further comparisons of existing frameworks based on different parameters are given in table 2, 3, 4, 5.

5 Open issues and challenges

In this section, some issues and challenges are highlighted that are provided direction to researchers for further research in this area.

1. *Standard protocol*
   MEC being a recent technology is evolving through the phases of implementation and requires standardization emanating from collaboration of industry and researchers over an agreed platform [12].
2. *Efficient Deployment*
   Minimizing the latencies through optimal utilization of bandwidth may be achieved with efficient deployment of MEC. However, it is difficult to optimize the spectrum usage with dependence on complex system components.
3. *User Mobility and Transparency*
   Provision of uninterrupted services to a frequently on-the-move client is another challenge in MEC environment with transparent process migration and platform heterogeneity.
4. *Heterogeneity and Scalability*
   As edge devices uses different access technologies including 3G, 4G, 5G, Wi-Fi and Wi-Max so aspect of heterogeneity should be catered in smooth functioning of MEC operations. This further necessitates the provision of scalability for different platforms with varying number of users [25, 77, 29, 36, 42, 41].
5. *Availability and Security*
   The availability of resources is mostly dependent upon server capacity and wireless access medium for ensuring constant service delivery. Along availability, security of data and applications from any intruder should be catered with physical measures.
6. *Fog-Cloud Interworking*
   When considering connectivity challenges for Gateways and/or Fog nodes, there are three different aspects to consider in any end-to-end system:
   - Northbound connections, which are the connections between Gateways/Fog nodes and a Cloud service (public or private).
   - Southbound connections, which are the connections between the Gateway/Fog node and the Edge devices/things/sensor networks.
   - East/West connections, which are the connections between Gateways/Fog nodes themselves, so that they can share data without requiring, Cloud connectivity.
7. *Data Management* The data management capabilities required include (but are not limited to):
   - Data normalization, which is ingesting, aligning and enriching the data from different sources (Things, devices and sensors) into a common data model with well understood semantics.
   - Filtering and querying data, so that applications and analytics can efficiently access and use the data relevant to them.
   - Integration with Edge analytics, because the whole reason for capturing these data is to be able to analyze them, create new actionable insights, make decisions and put those decisions into action.
   - Transforming data into different representations and formats, for the purposes of integrating with the IoT ecosystem.
   - Aggregating data and/or abstract meta-data, as preparation for local analytics or pushing it to Cloud services.

6 Conclusion

The paper presents a comprehensive review of the prevalent MEC frameworks along with a comparative analysis of contemporary approaches with respects to different performance parameters. Comparative analysis employs different parameters such as such as system performance, network performance, overhead of deployment and system migration overhead to measure the degree of effectiveness of different approaches. Based on our thorough investigation, it can be asserted that MEC is a way forward for achieving 1ms latency dream . Therefore, researcher has proposed several MEC architecture to reduce the latency. While considering the state-of-the-art presented in this paper, many areas are still open for further research to investigate a comprehensive architecture design with intelligent migration mechanism for multi-access mobile edge computing.

Declarations

Authors contributions

SS contributed to Cloudlet approach and Fog Consortium, classification and detail comparison of existing frameworks. MI has been involved in drafting the manuscript and revising it critically during initial and revised submissions. MI also contributed to the open issues and challenges. TD contributed to Edge Cloud Computing Standardization and Fora with a detailed comparison of edge cloud approaches including MEC approach. ZQ contributed to the introduction part. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no financial and non-financial competing interests.

Acknowledgment

We are thankful to the Sohail Sarwar for proofreading the paper to improve the write up of the manuscript.

Table 2 Comparison of Existing Frameworks Part-1

Properties	**[25]**	**[80]**	**[77]**	**[60]**	**[26]**	**[29]**	**[67]**	**[36]**	**[42]**
Improve execution cost	*Y*	*N*	*N*	*N*	*N*	*N*	*N*	*N*	*N*
Minimum execution time	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*Y*	*N*	*N*
Power consumption	*L*	*L*	*L*	*L*	*L*	*L*	*L*	*L*	*L*
Maximum resource utilization	*N/A*	*Y*	*N/A*	*N/A*	*N/A*	*N/A*	*N*	*N*	*N*
Caching support	*N/A*	*N/A*	*N/A*	*N*	*N*	*N*	*Y*	*N/A*	*N/A*
Scalability	*N*	*Y*	*N*	*Y*	*Y*	*N*	*Y*	*N*	*N*
Complexity	*N/A*	*N/A*	*N/A*	*L*	*H*	*L*	*L*	*L*	*N/A*
Augmentation of resource transparency	*N*	*N*	*N*	*N*	*Y*	*N*	*N/A*	*N/A*	*N/A*
Programmer support	*N*	*N*	*Y*	*N/A*	*N/A*	*Y*	*N/A*	*N*	*Y*
Parallel execution support	*N/A*	*N/A*	*N/A*	*N*	*N*	*N*	*N*	*Y*	*N/A*
Maximum throughput	*N*	*Y*	*N*	*N*	*N*	*N*	*N/A*	*N/A*	*N/A*
Network latency	*L*	*N*	*L*	*L*	*L*	*L*	*L*	*L*	*H*
Optimize bandwidth utilization	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N*	*N*	*N*
QoS	*N/A*	*N*	*Y*	*N*	*N/A*	*N*	*N/A*	*N/A*	*Y*
Guaranteed Bandwidth	*N/A*	*N/A*	*N/A*	*N*	*N*	*N*	*N*	*N/A*	*Y*
Network Load	*N/A*	*N/A*	*N/A*	*H*	*L*	*L*	*L*	*L*	*N/A*
Transmission delay	*L*	*H*	*H*	*N/A*	*N/A*	*L*	*N/A*	*M*	*L*
Reduction in number of hops	*N/A*	*N/A*	*N/A*	*Y*	*Y*	*Y*	*Y*	*N/A*	*N*
Security overhead	*M*	*N/A*	*N/A*	*N/A*	*M*	*N/A*	*M*	*N/A*	*H*
Fault tolerance	*N*	*N*	*Y*	*N/A*	*N/A*	*Y*	*N/A*	*N*	*Y*
Pre-execution delay	*N/A*	*N/A*	*N/A*	*H*	*H*	*H*	*M*	*N/A*	*N/A*
Usage of high bandwidth links	*N/A*	*N/A*	*N/A*	*Y*	*Y*	*Y*	*Y*	*N/A*	*Y*
Reduce offloading time	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N*	*N*	*Y*
Optimize data transfer cost	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*Y*	*Y*	*Y*

Y=Yes; N=No; H=High; M=Medium; L=Low; N/A=Not Applicable;

Table 3 Comparison of Existing Frameworks Part-2

Properties	**[25]**	**[80]**	**[77]**	**[60]**	**[26]**	**[29]**	**[67]**	**[36]**	**[42]**
Data transfer overhead	*H*	*H*	*H*	*N/A*	*H*	*L*	*L*	*H*	*M*
VM migration overhead	*H*	*N/A*	*N/A*	*N/A*	*H*	*N/A*	*L*	*N/A*	*M*
Optimize deployment	*N*	*N*	*Y*	*N*	*N*	*N*	*N/A*	*N/A*	*N/A*
Profiler Overhead	*H*	*M*	*L*	*N/A*	*L*	*H*	*N/A*	*H*	*L*
Cloud usage overhead	*H*	*M*	*L*	*L*	*H*	*L*	*M*	*L*	*L*
Operational cost	*H*	*L*	*L*	*M*	*H*	*L*	*M*	*L*	*L*
Deploys mirror	*N/A*	*N/A*	*N/A*	*N*	*N*	*N*	*N*	*N/A*	*N/A*
Partitioning overhead	*L*	*L*	*H*	*N/A*	*N/A*	*M*	*N/A*	*M*	*M*
Offloading overhead	*H*	*H*	*H*	*N/A*	*N/A*	*H*	*N/A*	*H*	*H*
Method call overhead	*H*	*H*	*H*	*N/A*	*N/A*	*H*	*N/A*	*H*	*H*

Y=Yes; N=No; H=High; M=Medium; L=Low; N/A=Not Applicable;

Table 4 Comparison of Existing Frameworks Part-3

Properties	**[64]**	**[11]**	**[76]**	**[41]**	**[83]**	**[37]**	**[52]**	**[45]**	**[69]**
Improve execution cost	*N*	*N*	*N*	*N*	*Y*	*N*	*N*	*N*	*N*
Minimum execution time	*N*	*N*	*N*	*N*	*N*	*Y*	*N*	*Y*	*N*
Power consumption	*L*	*L*	*L*	*L*	*L*	*L*	*L*	*L*	*M*
Maximum resource utilization	*N*	*N*	*Y*	*Y*	*N*	*N*	*N*	*N*	*N*
Caching support	*N/A*	*N/A*	*N*	*N*	*Y*	*Y*	*N/A*	*N/A*	*N/A*
Scalability	*Y*	*Y*	*Y*	*N*	*Y*	*Y*	*Y*	*Y*	*Y*
Programmer support	*N*	*N/A*	*Y*	*N*	*N/A*	*L*	*N/A*	*N/A*	*N/A*
Parallel execution support	*N/A*	*N/A*	*N*	*N*	*N*	*Y*	*N/A*	*N/A*	*N/A*
Network latency	*L*	*L*	*L*	*L*	*L*	*L*	*L*	*L*	*H*
Optimize bandwidth utilization	*Y*	*N*	*N*	*N*	*N*	*N*	*N*	*N*	*N*
QoS	*Y*	*N/A*	*Y*	*N*	*N/A*	*N/A*	*N*	*N/A*	*Y*
Minimum response time	*N*	*N*	*N*	*N*	*N*	*N*	*N*	*N*	*Y*
Guaranteed Bandwidth	*N/A*	*N/A*	*N*	*N*	*N*	*N*	*N/A*	*N/A*	*N/A*
Reduction in number of hops	*N/A*	*N/A*	*Y*	*Y*	*Y*	*Y*	*N/A*	*N/A*	*N/A*
Pre-execution delay	*N/A*	*N/A*	*M*	*H*	*L*	*L*	*N/A*	*N/A*	*N/A*
Usage of high bandwidth links	*N/A*	*N/A*	*Y*	*Y*	*N*	*Y*	*N/A*	*N/A*	*N/A*
Profiler Overhead	*N/A*	*N/A*	*H*	*N/A*	*N/A*	*N/A*	*Y*	*N/A*	*N/A*
Cloud usage overhead	*L*	*M*	*L*	*H*	*H*	*L*	*H*	*L*	*M*
Operational cost	*L*	*L*	*L*	*L*	*H*	*L*	*H*	*L*	*H*
Deploys mirror	*N/A*	*N/A*	*N*	*N*	*Y*	*N*	*N/A*	*N/A*	*N/A*

Y=Yes; N=No; H=High; M=Medium; L=Low; N/A=Not Applicable;

Table 5 Comparison of Existing Frameworks Part-4

Properties	**[75]**	**[82]**	**[48]**	**[49]**	**[33]**	**[50]**	**[51]**	**[38]**	**[34]**
Improve execution cost	*N*	*N*	*Y*	*N*	*N*	*N*	*Y*	*N/A*	*N/A*
Minimum execution time	*Y*	*N*	*Y*	*N*	*N*	*N*	*N*	*N/A*	*N/A*
Power consumption	*L*	*L*	*L*	*L*	*H*	*L*	*L*	*N/A*	*N/A*
Caching support	*Y*	*N/A*	*N*	*Y*	*N/A*	*N/A*	*Y*	*Y*	*Y*
Scalability	*N*	*Y*	*Y*	*Y*	*Y*	*Y*	*N*	*N/A*	*N/A*
Programmer support	*N*	*Y*	*Y*	*N/A*	*N/A*	*Y*	*Y*	*N/A*	*N/A*
Parallel execution support	*N/A*	*N/A*	*Y*	*N*	*N/A*	*N/A*	*N*	*N*	*N*
Maximum cache hit rate	*N/A*	*N*	*N*	*Y*	*N*	*N*	*N*	*N/A*	*N/A*
Minimum missed deadlines	*N/A*	*N*	*N*	*N*	*Y*	*N*	*N*	*N/A*	*N/A*
Network latency	*L*	*H*	*N/A*	*L*	*L*	*M*	*L*	*N/A*	*N/A*
QoS	*N*	*Y*	*N*	*N*	*N*	*Y*	*N*	*N/A*	*N/A*
Guaranteed Bandwidth	*N*	*N/A*	*N*	*N*	*N/A*	*N/A*	*N*	*N*	*N*
Transmission delay	*M*	*H*	*H*	*N/A*	*N/A*	*M*	*M*	*N/A*	*N/A*
Reduction in number of hops	*N*	*N/A*	*N*	*Y*	*N/A*	*N/A*	*Y*	*Y*	*Y*
Fault tolerance	*Y*	*Y*	*Y*	*N/A*	*N/A*	*N*	*N*	*N/A*	*N/A*
Pre-execution delay	*H*	*N/A*	*H*	*L*	*N/A*	*N/A*	*H*	*L*	*H*
Usage of high bandwidth links	*Y*	*N/A*	*Y*	*Y*	*N/A*	*N/A*	*Y*	*Y*	*Y*
Maximum privacy and security	*N/A*	*N*	*N*	*N*	*N*	*Y*	*Y*	*N/A*	*N/A*
Maximum throughput	*N/A*	*Y*	*N*	*N*	*N*	*N*	*N*	*N/A*	*Y*
Data transfer overhead	*H*	*L*	*H*	*N/A*	*N/A*	*H*	*L*	*H*	*N/A*
Profiler Overhead	*L*	*H*	*H*	*M*	*H*	*H*	*H*	*N/A*	*N/A*
Cloud usage overhead	*L*	*H*	*H*	*L*	*L*	*L*	*L*	*H*	*H*
Operational cost	*L*	*H*	*L*	*L*	*L*	*M*	*L*	*N/A*	*N/A*
Deploys mirror	*N*	*N/A*	*N*	*N*	*N/A*	*N/A*	*N*	*N*	*N*
Partitioning overhead	*H*	*H*	*H*	*N/A*	*N/A*	*H*	*L*	*N/A*	*N/A*
Offloading overhead	*H*	*H*	*H*	*N/A*	*N/A*	*L*	*H*	*N/A*	*N/A*
Method call overhead	*L*	*H*	*H*	*N/A*	*N/A*	*H*	*H*	*N/A*	*N/A*

Y=Yes; N=No; H=High; M=Medium; L=Low; N/A=Not Applicable;

References

1. Elijah: Cloudlet-based Edge Computing. `http://elijah.cs.cmu.edu/`. Accessed: 2017-07-19.
2. ETSI Mobile Edge Computing Portal. `http://www.etsi.org/technologies-clusters/technologies/mobile-edge-computing`. Accessed: 2016-01-27.
3. Internet of Everything. `http://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoE_Economy.pdf`. Accessed: 2017-07-17.
4. ITU-T report about Tactile Internet, Aug 2014. `https://www.itu.int/dms_pub/itu-t/oth/23/01/T23010000230001PDFE.pdf`. Accessed: 2016-01-27.
5. OpenEdgeComputing. `http://openedgecomputing.org/`. Accessed: 2017-07-19.
6. RCRWireless. `http://www.rcrwireless.com/20160628/opinion/reality-check-50b-iot-devices-connected-2020-beyond-hype-reality-tag10`. Accessed: 2017-07-17.
7. Second white paper on Mobile Edge Computing from ETSI. `http://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp11_mec_a_key_technology_towards_5g.pdf`. Accessed: 2016-01-27.
8. tactical-cloudlets. `https://www.govtechworks.com/tactical-cloudlets-mobile-computing-readies-for-battle/#gs.T=_DGTc`. Accessed: 2017-07-22.
9. Victor Bahl, Microsoft, 13 May 2015. `http://research.microsoft.com/en-us/um/people/bahl/Present/Bahl_mDC_Keynote_May.pdf`. Accessed: 2016-01-27.
10. Mohammad Aazam and Eui-Nam Huh. Fog computing and smart gateway based communication for cloud of things. In *Future Internet of Things and Cloud (FiCloud), 2014 International Conference on*, pages 464–470. IEEE, 2014.
11. Vamsi Subhash Achanta, Nishanth Talanki Sureshbabu, Varghese Thomas, M Lakshmi Sahitya, and Smitha Rao. Cloudlet-based multi-lingual dictionaries. In *Services in Emerging Markets (ICSEM), 2012 Third International Conference on*, pages 30–36. IEEE, 2012.
12. Arif Ahmed and Ejaz Ahmed. A survey on mobile edge computing. In *Intelligent Systems and Control (ISCO), 2016 10th International Conference on*, pages 1–8. IEEE, 2016.
13. Ejaz Ahmed, Abdullah Gani, Muhammad Khurram Khan, Rajkumar Buyya, and Samee U Khan. Seamless application execution in mobile cloud computing: Motivation, taxonomy, and open challenges. *Journal of Network and Computer Applications*, 52:154–172, 2015.
14. Ejaz Ahmed, Abdullah Gani, Mehdi Sookhak, Siti Hafizah Ab Hamid, and Feng Xia. Application optimization in mobile cloud computing: Motivation, taxonomies, and open challenges. *Journal of Network and Computer Applications*, 52:52–68, 2015.
15. Ejaz Ahmed and Mubashir Husain Rehmani. Mobile edge computing: opportunities, solutions, and challenges, 2017.
16. Michael Till Beck and Marco Maier. Mobile edge computing: Challenges for future virtual network embedding algorithms, 2014.
17. Michael Till Beck, Martin Werner, Sebastian Feld, and Thomas Schimper. Mobile edge computing: A taxonomy. 2014.
18. Flavio Bonomi, Rodolfo Milito, Preethi Natarajan, and Jiang Zhu. Fog computing: A platform for internet of things and analytics. In *Big Data and Internet of Things: A Roadmap for Smart Environments*, pages 169–186. Springer, 2014.
19. Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pages 13–16. ACM, 2012.
20. Alessio Botta, Walter De Donato, Valerio Persico, and Antonio Pescapé. On the integration of cloud computing and internet of things. In *Future Internet of Things and Cloud (FiCloud), 2014 International Conference on*, pages 23–30. IEEE, 2014.
21. Alberto Ceselli, Marco Premoli, and Stefano Secci. Cloudlet network design optimization. In *Proc. of IFIP Networking*, 2015.
22. Shanzhi Chen, Hui Xu, Dake Liu, Bo Hu, and Hucheng Wang. A vision of iot: Applications, challenges, and opportunities with china perspective. *IEEE Internet of Things journal*, 1(4):349–359, 2014.
23. Shuang Chen, Yanzhi Wang, and Massoud Pedram. A semi-markovian decision process based control method for offloading tasks from mobile devices to the cloud. In *Global Communications Conference (GLOBECOM), 2013 IEEE*, pages 2885–2890. IEEE, 2013.
24. Xu Chen, Lei Jiao, Wenzhong Li, and Xiaoming Fu. Efficient multi-user computation offloading for mobile-edge cloud computing. 2015.
25. Byung-Gon Chun, Sunghwan Ihm, Petros Maniatis, Mayur Naik, and Ashwin Patti. Clonecloud: elastic execution between mobile device and cloud. In *Proceedings of the sixth conference on Computer sys-*

tems, pages 301–314. ACM, 2011.

26. Byung-Gon Chun and Petros Maniatis. Augmented smartphone applications through clone cloud execution. In *HotOS*, volume 9, pages 8–11, 2009.
27. Sarah Clinch, Jan Harkes, Adrian Friday, Nigel Davies, and Mahadev Satyanarayanan. How close is close enough? understanding the role of cloudlets in supporting display appropriation by mobile users. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*, pages 122–127. IEEE, 2012.
28. Peter Corcoran and Soumya Kanti Datta. Mobile-edge computing and the internet of things for consumers: Extending cloud computing and services to the edge of the network. *IEEE Consumer Electronics Magazine*, 5(4):73–74, 2016.
29. Eduardo Cuervo, Aruna Balasubramanian, Dae-ki Cho, Alec Wolman, Stefan Saroiu, Ranveer Chandra, and Paramvir Bahl. Maui: making smartphones last longer with code offload. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, pages 49–62. ACM, 2010.
30. Amir Vahid Dastjerdi and Rajkumar Buyya. Fog computing: Helping the internet of things realize its potential. *Computer*, 49(8):112–116, 2016.
31. Mikael Desertot, Clement Escoffier, Philippe Lalanda, and Didier Donsez. Autonomic management of edge servers. In *Self-Organizing Systems*, pages 216–229. Springer, 2006.
32. Shuvashis Dey, Arjun Mukherjee, Himadri Sekhar Paul, and Arnab Pal. Challenges of using edge devices in iot computation grids. In *Parallel and Distributed Systems (ICPADS), 2013 International Conference on*, pages 564–569. IEEE, 2013.
33. Adam Dou, Vana Kalogeraki, Dimitrios Gunopulos, Taneli Mielikainen, and Ville H Tuulos. Misco: a mapreduce framework for mobile systems. In *Proceedings of the 3rd international conference on pervasive technologies related to assistive environments*, page 32. ACM, 2010.
34. Debessay Fesehaye, Yunlong Gao, Klara Nahrstedt, and Guijun Wang. Impact of cloudlets on interactive mobile cloud applications. In *Enterprise Distributed Object Computing Conference (EDOC), 2012 IEEE 16th International*, pages 123–132. IEEE, 2012.
35. Mohamed Firdhous, Osman Ghazali, and Suhaidi Hassan. Fog computing: Will it be the future of cloud computing? In *Third International Conference on Informatics & Applications, Kuala Terengganu, Malaysia*, pages 8–15, 2014.
36. Ioana Giurgiu, Oriana Riva, Dejan Juric, Ivan Krivulev, and Gustavo Alonso. Calling the cloud: enabling mobile phones as interfaces to cloud applications. In *Middleware 2009*, pages 83–102. Springer, 2009.
37. Mark S Gordon, Davoud Anoushe Jamshidi, Scott A Mahlke, Zhuoqing Morley Mao, and Xu Chen. Comet: Code offload by migrating execution transparently. In *OSDI*, pages 93–106, 2012.
38. Sachin Goyal and John Carter. A lightweight secure cyber foraging infrastructure for resource-constrained devices. In *Mobile Computing Systems and Applications, 2004. WMCSA 2004. Sixth IEEE Workshop on*, pages 186–195. IEEE, 2004.
39. Raffaella Grieco, Delfina Malandrino, and Vittorio Scarano. A scalable cluster-based infrastructure for edge-computing services. *World Wide Web*, 9(3):317–341, 2006.
40. Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.
41. Gonzalo Huerta-Canepa and Dongman Lee. A virtual cloud computing provider for mobile devices. In *Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond*, page 6. ACM, 2010.
42. Shih-Hao Hung, Chi-Sheng Shih, Jeng-Peng Shieh, Chen-Pang Lee, and Yi-Hsiang Huang. Executing mobile applications on the cloud: Framework and issues. *Computers & Mathematics with Applications*, 63(2):573–587, 2012.
43. Yaser Jararweh, Ahmad Doulat, Omar AlQudah, Ejaz Ahmed, Mahmoud Al-Ayyoub, and Elhadj Benkhelifa. The future of mobile cloud computing: integrating cloudlets and mobile edge computing. In *Telecommunications (ICT), 2016 23rd International Conference on*, pages 1–5. IEEE, 2016.
44. Yaser Jararweh, Loai Tawalbeh, Fadi Ababneh, and Fahd Dosari. Resource efficient mobile computing using cloudlet infrastructure. In *Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference on*, pages 373–377. IEEE, 2013.
45. Roelof Kemp, Nicholas Palmer, Thilo Kielmann, and Henri E Bal. Cuckoo: A computation offloading framework for smartphones. In *MobiCASE*, pages 59–79. Springer, 2010.
46. Kashif A Khan, Qi Wang, Chunbo Luo, Xinheng Wang, and Christos Grecos. Comparative study of internet cloud and cloudlet over wireless mesh networks for real-time applications. In *SPIE Photonics*

Europe, pages 91390K–91390K. International Society for Optics and Photonics, 2014.

47. Kashif Ali Khan, Qi Wang, and Christos Grecos. Experimental framework of integrated cloudlets and wireless mesh networks. In *Telecommunications Forum (TELFOR), 2012 20th*, pages 190–193. IEEE, 2012.
48. Sokol Kosta, Andrius Aucinas, Pan Hui, Richard Mortier, and Xinwen Zhang. Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading. In *INFOCOM, 2012 Proceedings IEEE*, pages 945–953. IEEE, 2012.
49. Emmanouil Koukoumidis, Dimitrios Lymberopoulos, Karin Strauss, Jie Liu, and Doug Burger. Pocket cloudlets. *ACM SIGARCH Computer Architecture News*, 39(1):171–184, 2011.
50. Dejan Kovachev, Yiwei Cao, and Ralf Klamma. Augmenting pervasive environments with an xmpp-based mobile cloud middleware. In *Mobile Computing, Applications, and Services*, pages 361–372. Springer, 2010.
51. Dejan Kovachev, Tian Yu, and Ralf Klamma. Adaptive computation offloading from mobile devices into the cloud. In *Parallel and Distributed Processing with Applications (ISPA), 2012 IEEE 10th International Symposium on*, pages 784–791. IEEE, 2012.
52. Byoung-Dai Lee. A framework for seamless execution of mobile applications in the cloud. In *Recent advances in computer science and information engineering*, pages 145–153. Springer, 2012.
53. Grace Lewis, Sebastián Echeverría, Soumya Simanta, Ben Bradshaw, and James Root. Tactical cloudlets: Moving cloud computing to the edge. In *Military Communications Conference (MILCOM), 2014 IEEE*, pages 1440–1446. IEEE, 2014.
54. Hongxing Li, Guochu Shou, Yihong Hu, and Zhigang Guo. Mobile edge computing: progress and challenges. In *Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2016 4th IEEE International Conference on*, pages 83–84. IEEE, 2016.
55. Kuan-Ching Li, Qing Li, and Timothy K Shih. *Cloud Computing and Digital Media: Fundamentals, Techniques, and Applications.* CRC Press, 2014.
56. Yujin Li and Wenye Wang. The unheralded power of cloudlet computing in the vicinity of mobile devices. In *Global Communications Conference (GLOBECOM), 2013 IEEE*, pages 4994–4999. IEEE, 2013.
57. Yanchen Liu, Myung J Lee, and Yanyan Zheng. Adaptive multi-resource allocation for cloudlet-based mobile cloud computing system. *IEEE Transactions on Mobile Computing*, 15(10):2398–2410, 2016.
58. Tom H Luan, Longxiang Gao, Zhi Li, Yang Xiang, and Limin Sun. Fog computing: Focusing on mobile users at the edge. *arXiv preprint arXiv:1502.01815*, 2015.
59. Andrew Machen, Shiqiang Wang, Kin K Leung, Bong Jun Ko, and Theodoros Salonidis. Migrating running applications across mobile edge clouds: poster. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 435–436. ACM, 2016.
60. Eugene E Marinelli. Hyrax: cloud computing on mobile devices using mapreduce. Technical report, DTIC Document, 2009.
61. Dusit Niyato, Ping Wang, Peter Chong Han Joo, Zhu Han, and Dong In Kim. Optimal energy management policy of a mobile cloudlet with wireless energy charging. In *Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on*, pages 728–733. IEEE, 2014.
62. M Patel, B Naughton, C Chan, N Sprecher, S Abeta, A Neal, et al. Mobile-edge computing introductory technical white paper. *White Paper, Mobile-edge Computing (MEC) industry initiative*, 2014.
63. Mugen Peng, Shi Yan, Kecheng Zhang, and Chonggang Wang. Fog computing based radio access networks: Issues and challenges. *arXiv preprint arXiv:1506.04233*, 2015.
64. Wang Qing, Hu Zheng, Wang Ming, and Liu Haifeng. Cactse: Cloudlet aided cooperative terminals service environment for mobile proximity content delivery. *Communications, China*, 10(6):47–59, 2013.
65. JeanMarc Rawadi, Hassan Artail, and Haidar Safa. Providing local cloud services to mobile devices with inter-cloudlet communication. In *Mediterranean Electrotechnical Conference (MELECON), 2014 17th IEEE*, pages 134–138. IEEE, 2014.
66. Hayat Routaib, Elarbi Badidi, Mouna Elmachkour, Essaid Sabir, and Mohammed Elkoutbi. Modeling and evaluating a cloudlet-based architecture for mobile cloud computing. In *Intelligent Systems: Theories and Applications (SITA-14), 2014 9th International Conference on*, pages 1–7. IEEE, 2014.
67. Mahadev Satyanarayanan, Paramvir Bahl, Ramón Caceres, and Nigel Davies. The case for vm-based cloudlets in mobile computing. *Pervasive Computing, IEEE*, 8(4):14–23, 2009.
68. Kshamata Shenoy, Prachi Bhokare, and Unnathi Pai. Fog computing future of cloud comput-

ing. *International Journal of Science and Research*, 4(6):55–56, 2015.

69. Tolga Soyata, Rajani Muraleedharan, Colin Funai, Minseok Kwon, and Wendi Heinzelman. Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture. In *Computers and Communications (ISCC), 2012 IEEE Symposium on*, pages 000059–000066. IEEE, 2012.
70. Ivan Stojmenovic and Sheng Wen. The fog computing paradigm: Scenarios and security issues. In *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*, pages 1–8. IEEE, 2014.
71. Rakesh Suryawanshi and GANESH MANDLIK. Focusing on mobile users at edge and internet of things using fog computing. 2015.
72. Tuyen X Tran, Abolfazl Hajisami, Parul Pandey, and Dario Pompili. Collaborative mobile edge computing in 5g networks: New paradigms, scenarios, and challenges. *IEEE Communications Magazine*, 55(4):54–61, 2017.
73. Nguyen B Truong, Gyu Myoung Lee, and Yacine Ghamri-Doudane. Software defined networking-based vehicular adhoc network with fog computing.
74. Luis M Vaquero and Luis Rodero-Merino. Finding your way in the fog: Towards a comprehensive definition of fog computing. *ACM SIGCOMM Computer Communication Review*, 44(5):27–32, 2014.
75. Tim Verbelen, Pieter Simoens, Filip De Turck, and Bart Dhoedt. Aiolos: Middleware for improving mobile application performance through cyber foraging. *Journal of Systems and Software*, 85(11):2629–2639, 2012.
76. Tim Verbelen, Pieter Simoens, Filip De Turck, and Bart Dhoedt. Cloudlets: bringing the cloud to the mobile user. In *Proceedings of the third ACM workshop on Mobile cloud computing and services*, pages 29–36. ACM, 2012.
77. Tim Verbelen, Tim Stevens, Pieter Simoens, Filip De Turck, and Bart Dhoedt. Dynamic deployment and quality adaptation for mobile augmented reality applications. *Journal of Systems and Software*, 84(11):1871–1882, 2011.
78. Xianbin Wang, Guangjie Han, Xiaojiang Du, and Joel JPC Rodrigues. Mobile cloud computing in 5g: Emerging trends, issues, and challenges [guest editorial]. *IEEE Network*, 29(2):4–5, 2015.
79. Md Whaiduzzaman, Abdullah Gani, and Anjum Naveed. Pefc: Performance enhancement framework for cloudlet in mobile cloud computing. In *Robotics and Manufacturing Automation (ROMA), 2014 IEEE International Symposium on*, pages 224–229. IEEE, 2014.
80. Lei Yang, Jiannong Cao, Yin Yuan, Tao Li, Andy Han, and Alvin Chan. A framework for partitioning and execution of data stream applications in mobile cloud computing. *ACM Sigmetrics Performance Evaluation Review*, 40(4):23–32, 2013.
81. Ke Zhang, Yuming Mao, Supeng Leng, Yejun He, and Yan ZHANG. Predictive offloading in cloud-driven vehicles: Using mobile-edge computing for a promising network paradigm. *IEEE Vehicular Technology Magazine*, 2017.
82. Xinwen Zhang, Sangoh Jeong, Anugeetha Kunjithapatham, and Simon Gibbs. Towards an elastic application model for augmenting computing capabilities of mobile platforms. In *Mobile wireless middleware, operating systems, and applications*, pages 161–174. Springer, 2010.
83. Bo Zhao, Zhi Xu, Caixia Chi, Sencun Zhu, and Guohong Cao. Mirroring smartphones for good: A feasibility study. In *Mobile and ubiquitous systems: computing, networking, and services*, pages 26–38. Springer, 2012.
84. Wenwu Zhu, Chong Luo, Jianfeng Wang, and Shipeng Li. Multimedia cloud computing. *Signal Processing Magazine, IEEE*, 28(3):59–69, 2011.
85. Weihua Zhuang. Emerging technologies, applications, and standardizations for connecting vehicles (part ii)[from the guest editor]. *IEEE Vehicular Technology Magazine*, 12(2):23–25, 2017.