Digital Communications and Networks 4 (2018) 77–86

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/en/journals/digital-communications-and-networks/

Edge computing technologies for Internet of Things: a primer

Yuan Ai, Mugen Peng *, Kecheng Zhang

Key Laboratory of Universal Wireless Communications for Ministry of Education, Beijing University of Posts and Telecommunications, China

A R T I C L E I N F O

Keywords:
Internet of Things (IoT)
Mobile edge computing
Cloudlets
Fog computing

A B S T R A C T

With the rapid development of mobile internet and Internet of Things applications, the conventional centralized cloud computing is encountering severe challenges, such as high latency, low Spectral Efficiency (SE), and non-adaptive machine type of communication. Motivated to solve these challenges, a new technology is driving a trend that shifts the function of centralized cloud computing to edge devices of networks. Several edge computing technologies originating from different backgrounds to decrease latency, improve SE, and support the massive machine type of communication have been emerging. This paper comprehensively presents a tutorial on three typical edge computing technologies, namely mobile edge computing, cloudlets, and fog computing. In particular, the standardization efforts, principles, architectures, and applications of these three technologies are summarized and compared. From the viewpoint of radio access network, the differences between mobile edge computing and fog computing are highlighted, and the characteristics of fog computing-based radio access network are discussed. Finally, open issues and future research directions are identified as well.

1. Introduction

Over the past decades, cloud computing has been greatly developed and applied owing to its high cost-efficiency and flexibility achieved through consolidation, in which computing, storage, and network management functions work in a centralized manner. With the rapid development of mobile internet and Internet of Things (IoT) applications, the existing centralized cloud computing architecture is encountering severe challenges. Mobile devices connected to distant centralized cloud servers try to obtain sophisticated applications, which impose additional load on both Radio Access Networks (RANs) and backhaul networks and result in high latency [1]. In addition, with the explosive growth in various access devices and end-user demands, IoT is driving a digital transformation in all aspects of the current modern life [2]. It is estimated by Cisco that the number of devices connected to IoT will become 50 billion by 2020 [3]. The emerging IoT introduces new challenges, such as stringent latency, capacity constraints, resource-constrained devices, uninterrupted services with intermittent connectivity, and enhanced security, which cannot be adequately addressed by the centralized cloud computing architecture [4]. An advanced cloud computing paradigm that breaks through the centralized architecture and alleviates the capacity and latency constraints is urgently required to cope with these challenges.

IoT refers to the interaction and communication between billions of devices that produce and exchange data related to real-world objects (i.e., things) [5]. IoT's features, including ultra-largescale network of things, device and network level heterogeneity, and large numbers of events generated by these things, will make the development of diverse applications and services a very challenging task [6]. These requirements are becoming difficult to accomplish in the IoT+ cloud scenario. IoT applications generate enormous amounts of data by IoT sensors. Big data are subsequently analyzed to determine reactions to events or to extract analytics or statistics. However, sending all the data to the cloud will require prohibitively high network bandwidth. Recent research efforts are investigating on how to effectively exploit capabilities at the edge of networks to support the IoT and its requirements [7]. In edge computing, the massive data generated by different types of IoT devices can be processed at the network edge instead of transmitting them to the centralized cloud infrastructure owing to bandwidth and energy consumption concerns. Edge computing can provide services with faster response and greater quality, in comparison with cloud computing. Edge computing is more suitable to be integrated with IoT to provide efficient and secure services for a large number of end-users, and edge computing-based architecture can be considered for the future IoT infrastructure [8].

Recently, nascent technologies and applications are driving a trend in the computing and communication landscape that shifts the function of centralized cloud computing into the edge devices of networks [9]. Software Defined Networking (SDN) and the associated concept of

* Corresponding author.
E-mail addresses: aiyuan@bupt.edu.cn (Y. Ai), pmg@bupt.edu.cn (M. Peng), buptzkc@163.com (K. Zhang).

https://doi.org/10.1016/j.dcan.2017.07.001
Received 24 April 2017; Received in revised form 28 June 2017; Accepted 2 July 2017
Available online 8 July 2017

Network Function Virtualization (NFV) are proposed as emerging solutions for future networks [10]. In particular, NFV enables edge devices to provide computing services and operate network functions by creating multiple Virtual Machines (VMs). Moreover, ultra-low latency is identified as one of the major requirements of 5th Generation (5G) RANs [11]. To decrease the latency, mobile operators are prone to deploying the application and content at the edge of networks. Meanwhile, operators can open the edge devices of RANs to third-party partners, allowing them to rapidly deploy innovative applications and content toward mobile subscribers, enterprisers, and other vertical segments [12]. Although the computing capabilities of wearable watches, smart phones, and other IoT devices have been significantly improved, they are still constrained by the fundamental challenges, such as memory size, battery life, and heat dissipation. Mobile devices need to extend battery lifetime by offloading energy-consuming computation of applications to the edge of networks [13].

Motivated to efficiently provide the massive machine type of communication, ultra-reliable low-latency communication, and high Spectral Efficiency (SE), industry investment and research interest focused on the edge of networks have grown dramatically [14]. To support low-latency requirements for resource-intensive applications, a new architectural element called cloudlets has been proposed in Ref. [15]. In order to accelerate the development of the ecosystem based on cloudlets, the Open Edge Computing (OEC) initiative has been launched in June 2015 by Vodafone, Intel, and Huawei companies in partnership with Carnegie Mellon University. Similarly, Nokia Networks company introduced a computing platform in 2013, which is integrated to the base station [16]. The original concept that applications and services are executed at the edge of networks has evolved gradually. In September 2014, a new Industry Specification Group (ISG) was proposed to be set up in European Telecommunications Standards Institute (ETSI) to allow the creation of industry specifications for Mobile Edge Computing (MEC), which has been supported by Huawei, IBM, Intel, Nokia Networks, NTT DoCoMo, Vodafone, and other companies [17]. In the MEC World Congress 2016, MEC ISG has renamed mobile edge computing as multi-access edge computing in order to reflect the growing interests from non-cellular operators. It is anticipated that multi-access edge computing will take effect starting from the end of 2017 [18]. Another technology similar to edge computing is known as fog computing, which was initiated by Cisco in 2012 [19]. To accelerate the adoption of fog computing, the OpenFog Consortium has been founded by ARM, Cisco, Dell, Intel, Microsoft and Princeton University in November 2015.

As there are three typical edge computing solutions for IoT, it is necessary to highlight the differences among them, and the development of a framework for edge computing technologies with reference to background, system architecture, and key techniques is challenging. The existing tutorial articles on edge computing are presented in Table 1. The scope of these existing publications provided neither an in-depth discussion nor a comparison in terms of improving SE, decreasing latency, and supporting the massive machine type of communication from the perspectives of the overall system architecture and key techniques. It is critical, therefore, to show a tutorial on edge computing by presenting a comprehensive review framework. Considering the important aspects and active research activities of edge computing, a tutorial of system architectures, key techniques, and application characteristics is presented in this paper to promote research and commercial success of edge computing. Additionally, given the extensiveness of the research area, more open issues and challenging works on both transforming the well-established infrastructure of fog computing for 5G RANs that are necessary to perform elaborative investigations in the future, are introduced as well.

The rest of this paper is organized as follows: Section 2 presents a review on the principles and applications of cloudlets. The standardization efforts, applications, architecture, and key technologies of mobile edge computing are presented in Section 3. Section 4 summarizes the standardization efforts, applications, and system architecture of fog computing, and compares MEC and fog computing-based RANs (F-RANs). The open issues and challenges are discussed in Section 5, prior to the conclusion in Section 6.

Table 1
Summary of existing survey articles on edge computing.

Aspects	Survey papers	Contributions
Cloudlet	[15]	A new cloudlet-based architecture for overcoming the technical obstacles in mobile computing.
	[24]	An open ecosystem based on the concept of cloudlets supporting many exciting mobile applications.
Mobile edge computing	[12]	An overview of MEC definition, architectural blueprint, requirements, and challenges of MEC as well as the objectives of the MEC initiative.
	[13]	A comprehensive survey of the state-of-the-art MEC research focusing on joint radio and computational resource management.
	[41]	A comprehensive survey of major use cases and reference scenarios, current advancement in standardization of MEC, and research on computation offloading.
Fog computing	[4]	A summary of the opportunities and challenges of fog computing focusing primarily on the networking context of IoT.
	[54]	An overview of Fog computing definition, reference architecture, use cases and challenges for fog computing as well as the future research and work.
Comprehensive surveys	[9]	An overview of edge computing definition, origin and background, challenges, and applications. Discussions of the future research directions of edge computing.
	[14]	A comprehensive comparison of three approaches: fog computing, MEC and Cloudlet. Discussions of further work and research in order to get concepts like Fog, MEC and Cloudlets adopted by industry.
	This article	A comprehensive tutorial of three state-of-the-art edge computing technologies, namely MEC, cloudlets, and fog computing. A comparison of standardization efforts, principles, architectures, and applications for these three technologies. The difference between mobile edge computing and fog computing from the viewpoint of RANs.

2. Cloudlet: the edge of internet

One of the critical challenges in cloud computing is the end-to-end responsiveness between a mobile device and associated cloud [20]. To address this challenge, the cloudlet, which is a mobility-enhanced small-scale cloud Data Center (DC) that is located at the edge of the internet, is proposed. A cloudlet is a trusted, resource-rich computer or cluster of computers that are well-connected to the internet and available for use by nearby mobile devices [15].

The main purpose of the cloudlet is supporting resource-intensive and interactive mobile applications by providing powerful computing resources to mobile devices with lower latency. User Equipments (UEs) can access the computing resources in the nearby cloudlet through a one-hop high-speed wireless local area network. Cloudlets represent the middle tier of the 3-tier hierarchy architecture (mobile device layer, cloudlet layer, and cloud layer) to achieve crisp response time. There are few but important differentiators between cloud and cloudlet:

1) Compared to the cloud DC, a cloudlet needs to be much more agile in its provisioning because the association with mobile devices is highly dynamic with considerable churn due to user mobility;

2) To support user mobility, a VM handoff technology needs to be used to seamlessly migrate the offloaded services on the first cloudlet to the second cloudlet as a user moves away from the currently associated cloudlet;
3) Because cloudlets are small DCs distributed geographically, a mobile device first has to discover, select, and associate with the appropriate cloudlet among multiple candidates before it starts provisioning.

2.1. Principle of cloudlet

Mobile users exploit the VM technology to rapidly instantiate a customized software on a nearby cloudlet. Generally, the customization of a base system specialized for a certain application is small. If the base VM exists on the cloudlet, only its difference relative to the desired custom VM, called a VM overlay, needs to be transferred. The approach of using VM overlays to provisioning cloudlets is called VM synthesis [21]. Cloudlets use the approach of VM synthesis for rapid provisioning and VM handoff. Fig. 1 shows the relevant steps of dynamic VM synthesis. A mobile device delivers a small VM overlay to the cloudlet that already possesses the base VM from which the overlay is derived (the delivery can be either from the cloud or from the storage on the mobile device). The cloudlet decompresses the overlay, applies it to the base VM to derive the launched VM, and then creates a VM instance from it. The mobile device can begin performing offload operations at this instance. At the end of the session, the instance is destroyed, but the launched VM image can be retained in a persistent cache for future sessions. To retain some training data for future offload sessions, the cloudlet generates a VM residue that can be sent back to the mobile device and incorporated into its overlay. Experimental results show that cloudlets can decrease the response time by 51% and reduce energy consumption by up to 42% in a mobile device compared to cloud offload [22]. The VM handoff is responsible for seamlessly transferring VM-encapsulated execution to an optimal offload site as users move. Validation experiments confirm that the resulting mechanism is a promising technique for enabling user mobility with low end-to-end latency applications [23].

2.2. Application of cloudlet

An open ecosystem based on the concept of cloudlets supports and enables many exciting mobile applications that are both compute-intensive and latency-sensitive [24]. For example, by leveraging a low end-to-end latency, the real-time interaction can be implemented on wearable cognitive assistance [25]. By real-time data analysis at the edge of the internet, cloudlets can reduce ingress bandwidth into the cloud [26]. By serving as physically proximate representatives of the cloud that are unavailable owing to failures or cyberattacks, cloudlets can improve robustness and availability in hostile environment [27]. Moreover, cloudlets can enable mobile access to the huge legacy world of Windows-based desktop applications. A VM encapsulating the personal desktop environment of a user is run on a cloudlet, and the user connects to it through a remote desktop protocol. Thus, users can use Windows-based desktop applications on a mobile device such as an Android tablet that is typically ARM-based.

To drive the development of software ecosystems surrounding cloudlets, the OEC initiative has been launched, synchronizing work with other efforts by ETSI ISG MEC and OPNFV. In addition, Carnegie Mellon University has implemented an open source platform called Open-Stack++ that is a derivative of the widely used OpenStack platform for cloud computing, which extends the functionality of Openstack to support cloudlets. Some key technologies such as cloudlet discovery, just-in-time provisioning and VM hand-off, have been implemented to be available as open source [28].

3. Mobile edge computing: the edge of mobile network

MEC is identified as a key enabler for IoT and for mission-critical, vertical solutions, and is recognized as one of the key architectural concepts and technologies. The concept of MEC was defined by ETSI as a new technology that provides an IT service environment and cloud-computing capabilities at the edge of the mobile network, within the RAN and in close proximity to mobile subscribers [12]. ETSI has published a white paper on MEC, and MEC has been considered a key emerging technology, which is important for future generation networks [17].

3.1. MEC standardization

ETSI has established the ISG on MEC to develop a standardized, open environment that will allow efficient and seamless integration of third-party applications across multi-vendor platforms in December 2014. Until January 2017, MEC ISG has released six specifications, one of which provides a glossary of terms related to the conceptual, architectural, and functional elements of MEC [29]. The purpose of that specification is to enable the consistent use of terminology within ETSI MEC specifications and, beyond the ISG, more widely in the industry. Another specification provides technical requirements enabling interoperability and deployment and describes example use cases and their technical benefits [30]. The other specification provides a framework and reference architecture to enable mobile edge applications to run efficiently

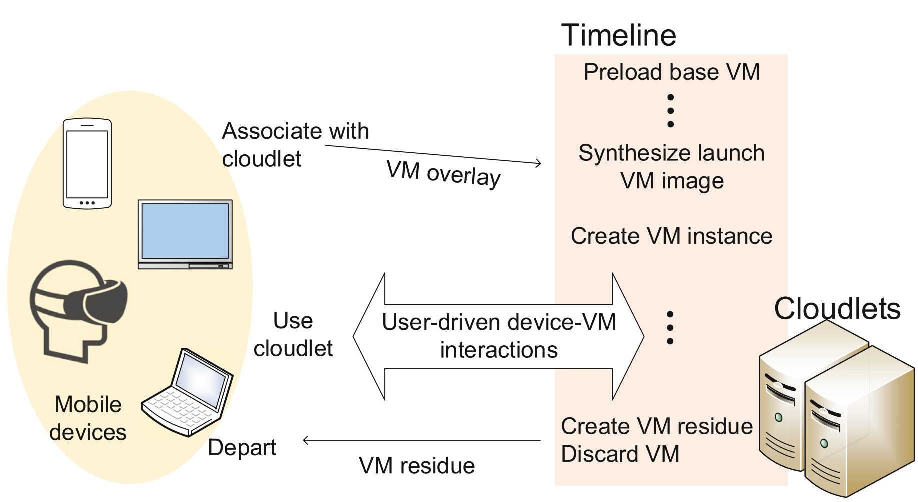

Fig. 1. Dynamic VM synthesis.

and seamlessly in a mobile network [31]. Moreover, the fourth specification in MEC ISG introduces a number of service scenarios that would benefit from the MEC technology [32]. The Proof of Concept (PoC) framework specification defines a framework to coordinate and promote multi-vendor PoC projects illustrating key aspects of MEC technology [33]. PoCs are essential to demonstrate MEC as feasible and valuable, to validate the specifications that are being developed, to demonstrate use cases, and ultimately to help develop a diverse and open MEC ecosystem. The last specification describes various metrics, which can potentially be improved through deploying a service on a MEC platform, such as latency, energy efficiency, network throughput, system resource footprint, and quality [34]. Furthermore, the last specification also describes the best practices for measuring such performance metrics. ETSI has announced six different MEC PoCs in September 2016, which have been accepted in MEC World Congress in Munich and have contributed to strengthen the strategic planning and decision-making of organizations, and help to identify which MEC solutions may be viable in the network. MEC ISG is now working on 9 new studies related to MEC APIs, management interfaces, and essential platform functionality. In addition, the MEC in an NFV environment is emerging on an end-to-end mobility. The NFV platform may be dedicated to MEC or shared with other network functions or applications. MEC exploits the NFV management and orchestration entities and interfaces as much as possible.

3.2. Application of MEC

Owing to its advanced features, such as low latency, proximity, high bandwidth, and real-time insight into radio network information and location awareness, MEC enables a large number of new types of applications and services for multiple sectors, such as consumer, enterprise, and health. In particular, MEC is deemed to be a promising solution for handling video streaming services in the context of smart cities [35]. Video streams from monitoring devices are locally processed and analyzed at a MEC server to extract meaningful data from video streams. The valuable data can be transmitted to the application server to reduce core network traffic. Augmented Reality (AR) mobile applications have inherent collaborative properties in terms of data collection in the uplink, computing at the edge, and data delivery in the downlink [36]. AR data require low latency and a high rate of data processing in order to provide the correct information depending on the location of the user. The processing of data can be performed on a local MEC server rather than on a centralized server to provide a perfect user experience. In addition, a vehicular delay-tolerant network-based smart grid data management scheme that leverages the mobile edge computing paradigm was proposed in Ref. [37]. The IoT generates additional messaging on telecommunication networks, and requires gateways to aggregate the messages and ensure low latency and security. A new architecture by leveraging MEC to collect, classify, and analyze the IoT data streams was introduced in Ref. [38]. The MEC server is responsible for managing various protocols, distribution of messages, and for processing of analytics. The MEC environment creates a new value chain and an energized ecosystem, which in turn creates new opportunities for mobile operators and application and content providers.

3.3. System architecture of MEC

As shown in Fig. 2, the MEC reference architecture, as described by ETSI [31], enables the implementation of MEC applications as software-only entities that run on the MEC host. The mobile edge platform offers the essential environment and functionality required to run the MEC application. MEC applications are running as VM on top of the virtualization infrastructure, and can interact with the mobile edge platform to perform certain support procedures related to the life-cycle of the application. Furthermore, the virtualization infrastructure includes a data plane that executes the traffic rules received by the mobile edge platform and routes the traffic among applications, local networks, and external networks. The MEC host level management comprises the mobile edge platform manager and the virtualization infrastructure manager. The former manages the life cycle of applications and the application rules and requirements including service authorizations, traffic rules, DNS configuration, and resolving conflicts. The latter is responsible for allocating, managing, and releasing visualized (compute, storage, and networking) resources of the virtualization infrastructure.

The operations support system receives request by a user application via a life-cycle management proxy, or by operators' third-party customers via the customer-facing service portal. The operation support system

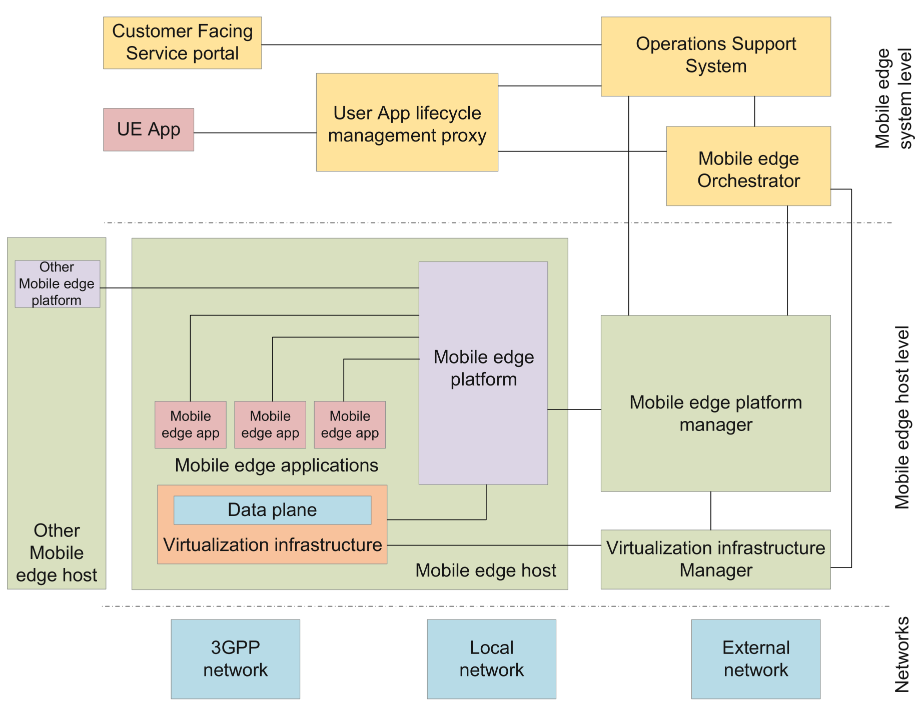

Fig. 2. Mobile edge system reference architecture.

decides whether the requests are granted or not. The granted requests are forwarded to the MEC orchestrator for further processing. The MEC orchestrator is the core functionality, as it maintains an overall view based on the deployed MEC hosts, available resources, available MEC services, and topology. For reasons of performance, costs, scalability, and operator preferred deployments, MEC supports different deployment scenarios [30], such as at the cellular macro base station (eNodeB) site, at the 3G Radio Network Controller (RNC) site, at a multi-radio access technology cell aggregation site, and at an aggregation point (which may also be at the edge of the core network, e.g., in a distributed DC). A network planning problem on determining where to optimally install the MEC servers among the available sites to find a tradeoff between installation costs and Quality of Service (QoS) has been explored in Ref. [39].

3.4. Key technologies of MEC

The key technologies of MEC including computation offloading and mobility management are illustrated in Fig. 3. Computation offloading is a procedure that migrates resource-intensive computations from a mobile device to the resource-rich nearby infrastructure [40]. Although mobile devices are constrained by computing capabilities, battery life, and heat dissipation, MEC enables running new sophisticated applications at UEs by offloading energy-consuming computations of the applications to the MEC server. An important part of computation offloading is to decide whether to offload or not, whether full or partial offloading is applicable, and what and how the computation could be offloaded. The offloading decision depends on the application model, which can be classified according to three criteria [41]. The first criterion is whether the application contains non-offloadable parts that cannot be offloaded (e.g., user input, camera, or acquired position that need to be executed at UEs). Second, there is no way to estimate the amount of data to be processed for some continuous-execution applications. The third criterion is a mutual dependency of individual parts to be processed. Generally, UE needs to be composed of a code profiler, system profilers, and decision engine to manage the offloading process [42]. The code profiler is responsible for managing what to offload depending on the application type and code/-data partitioned. System profilers are in charge of monitoring multiple parameters, such as available bandwidth, data size to transmit, and energy to execute the code. These parameters influence when to offload. The decision engine determines whether to offload or not. Computation offloading decision algorithms have been investigated and compared comprehensively in Ref. [41]. The majority of algorithms aim to minimize the energy consumption at the mobile device, subject to the execution delay acceptable by the offloaded application, or to find an optimal tradeoff between these two metrics. Numerical results demonstrate that MEC can improve energy efficiency by computation offloading in heterogeneous networks [43]. A game theoretic approach was proposed for a computation offloading decision-making problem among multiple mobile device users in Ref. [44]. Numerical results show that the proposed algorithm achieves an excellent computation offloading performance and scales well as the user size increases. The energy-efficient resource allocation problems for computation offloading have been researched in Ref. [45]. In addition, some efforts have been focused on the joint optimization of radio and computational resources, aimed at minimizing energy consumption under latency and power budget constraints [46].

When the UE performs a handover to another cell, it is important to guarantee the service continuity and QoS requirements [47]. Several MEC applications expect to continue serving the UE after a location change. If the UE forwards the computation to the MEC, VM migration needs to be initiated to provide continuity of the service. VM migration is a service included in many hyper-visors to move VMs from one physical machine to another, commonly within a DC [48]. A crucial part is to decide whether VM migration depends on the service type and requirements, content size, and user class [49]. Note that an ongoing video service with strict QoS requirements may be migrated, and delay-sensitive measurement tasks for an emergency warning machine type communication services are always migrated to the optimal MEC server.

On one hand, a decision must be made indicating whether the service must be fully or partially migrated, while considering the VM migration cost including the time required for the VM migration and backhaul resources consumption because of traffic exchanges among computing nodes. On the other hand, the benefit of VM migration is to reduce the service delay. Meanwhile, backhaul resources do not have to be allocated for the transmission of computation results back to the UE. A profit maximization avatar placement strategy was proposed in order to optimize the tradeoff between the migration gain and the migration cost by selectively migrating the VM to their optimal locations [50]. In addition, a prediction technology used for the dynamic VM placement and to find the most suitable communication path according to the expected users' movement has been explored in Ref. [51]. Comparing to state-of-the-art approaches, the proposed algorithm reduces the offloading delay by a value between 10% and 66%.

4. Fog computing: edge work with cloud

The OpenFog Consortium was founded to drive industry and academic leadership in fog computing architecture, testbed development, and a variety of inter-operability and composability deliverables that seamlessly leverage cloud and edge architectures to enable end-to-end IoT scenarios [52]. OpenFog Consortium published a white paper on fog computing in February 2016, in which the consortium's approach to an open fog computing architecture (OpenFog architecture) has been outlined [53]. The OpenFog Consortium defines fog computing as a system-level horizontal architecture that distributes resources and services of computing, storage, control and networking anywhere along the continuum from the cloud to things. Fog computing is different from edge computing and provides tools for distributing, orchestrating, managing, and securing resources and services across networks and between devices that reside at the edge. Edge architecture places servers, applications, and

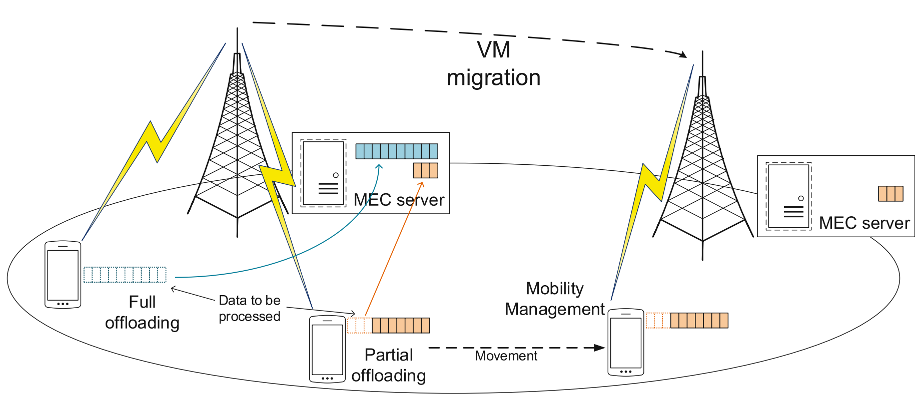

Fig. 3. Offloading and mobility management.

small clouds at the edge. Fog jointly works with the cloud, while edge is defined by the exclusion of cloud.

4.1. Standardization of fog computing

The fog computing standardization is mainly charged by the OpenFog Consortium, whose objective is to influence standards bodies to create standards so that IoT systems at the edge can inter-operate securely with other edge and cloud services in a friction-free environment. The OpenFog Consortium has set up six working groups, namely architecture working group, communications working group, manageability working group, security working group, software infrastructure working group, and testbed working group. These working groups evaluate, classify, and recommend standards, practices, and technologies that are appropriate for the OpenFog architecture to address corresponding challenges.

The OpenFog Consortium announces the release of the OpenFog Reference Architecture in February 2017, which is a universal technical framework designed to enable the data-intensive requirements of IoT, 5G, and Artificial Intelligence (AI) applications [54]. This architecture is the baseline to develop an open architecture fog computing environment, which creates a roadmap and is the first step in creating standards for fog computing. The OpenFog Consortium will establish detailed guidance, interface with standards organizations such as IEEE on recommending standards, and specifying APIs for key interfaces in the reference architecture.

The structural aspects and perspectives of the reference architecture, which is used as a common baseline for achieving a multi-vendor interoperable fog computing ecosystem, are illustrated in Fig. 4. It is a composite of multiple views to address stake-holders in the fog computing value chain, such as software view, system view, and node view. The node view is the lowest level view, which includes the protocol abstraction layer and sensors, actuators, and control. The system view is composed of one or more node views coupled with other components to create a platform. Software view comprises the top three layers that sit on top of the platform hardware layer. The software is running on fog platforms to satisfy a use case requirement. Five cross-cutting perspectives are employed throughout fog computing implementations, namely (1) performance and scale perspective, (2) security perspective, (3) manageability perspective, (4) data analytic and control perspective, and (5) IT business and cross-fog applications perspective.

4.2. Application of fog computing

An open architecture based on fog computing enable interoperability in IoT, 5G, AI, tactile internet, virtual reality, and other complex data and network intensive applications. IoT applications generate unprecedented amounts of data that can be useful in many ways [55]. Based on this situation, fog nodes can be used to carry out data mining and data analysis on a large volume of multi-modal and heterogeneous data from various sensor devices and other IoT devices to achieve real time and fast processing for decision making [56]. A hierarchical fog computing architecture for big data analysis in smart cities was introduced in Ref. [57]. Meanwhile, a prototypical system for smart pipeline monitoring was constructed to experimentally evaluate the performance of this architecture. Experimental results demonstrate the feasibility of the system's city-wide implementation in future smart cities scenario. A privacy-preserving protocol for enhancing security in vehicular crowd sensing-based road surface condition monitoring system using fog computing was proposed in Ref. [58]. The fog computing-based face identification and resolution framework have been explored to solve some security and privacy issues [59]. In addition, because the fog is localized, new services that require mobile networks supporting high data rates and low latency become possible, such as virtual reality. The vehicle can be employed as the fog node to attain optimum utilization of these vehicular communications and computational resources [60]. The mobile fog node can communicate with other fog nodes or provide services including infotainment, advanced driver assistance systems, autonomous driving, collision avoidance, and navigation. Emergency, health care, and other latency-sensitive and security-/privacy-sensitive services require fog nodes to be executed between the underlying nodes and the distant cloud [61]. Extensive experimental results validate that fog computing supporting medical cyber-physical system can improve the cost efficiency significantly by jointly considering base station association, task distribution, and VM deployment [62]. Fog computing provides business value for some applications that require real-time decision making, low latency, improved security, and are network-constrained.

4.3. System architecture of fog computing

A typical hierarchical architecture based on fog computing is shown in Fig. 5. From the functional point of view, a fog node has several functions, including networking, computing, accelerating, storing, and

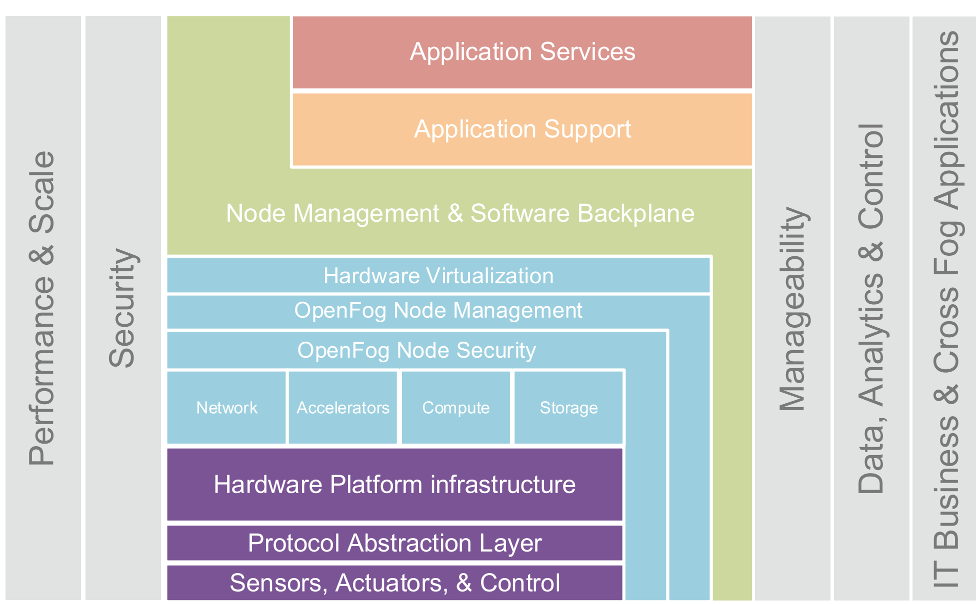

Fig. 4. The OpenFog reference architecture description with perspectives.

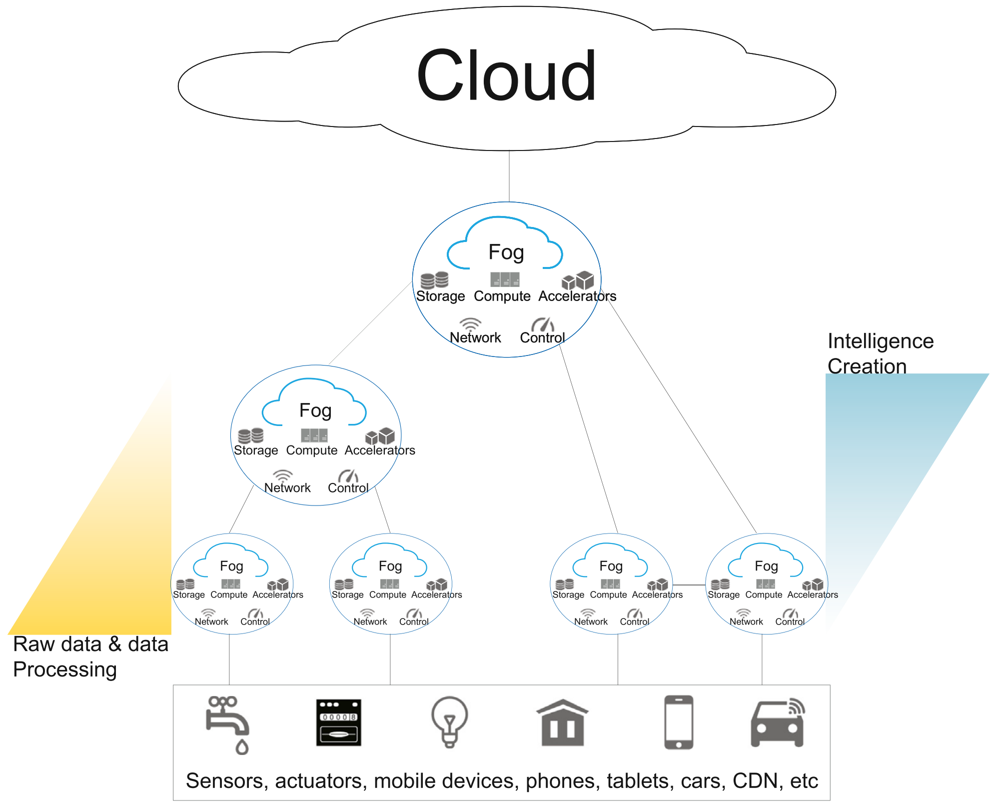

Fig. 5. A typical hierarchical architecture based on fog computing.

control. Fog nodes can communicate with each other through wired or wireless transmission. Moreover, fog nodes have some general computing capabilities. In particular, those fog nodes engaged in enhanced analytics need to configure accelerator modules such as graphics processing units, field programmable gate arrays, and digital signal processors to provide supplementary computational throughput. Many types of storage are required in fog nodes to meet the required reliability and data integrity of the system and scenario. Generally, there are a rich set of sensors and actuators at the edge of the network in an application scenario. These sensors and actuators are connected to the fog node via a multitude of interfaces, such as PCIe, USB, and Ethernet. Fog nodes can be worked in a mesh manner to provide load balancing, resilience, fault tolerance, data sharing, and minimization of cloud communication. There are often three tiers in a fog computing system, but more tiers can be allowed for a special application scenario. At the edge of the network, fog nodes are typically focused on sensor data acquisition/collection, data normalization, and command/control of sensors and actuators. At the next higher tier, fog nodes are focused on data filtering, compression, and transformation. At the higher tiers or nearest the backend cloud, fog nodes are focused on aggregating data and turning the data into knowledge. Architecturally, fog nodes at the edge may require less processing, communications, and storage than nodes at high levels. However, Input and Output (I/O) accelerators required to facilitate sensor data intake at the edge are much larger in aggregate than I/O accelerators designed for higher-level nodes. With the increase in the number of tiers, each tier would be sifting and extracting meaningful data to create more intelligence.

The conventional centralized cloud computing continues to remain an important part of computing systems as fog computing emerges. Cloud and edge computing complement each other to form a mutually beneficial and inter-dependent service continuum. Some functions are naturally more advantageous to carry out in centralized cloud, while others are better suited at the edge. In Ref. [63], a quantitative analysis of energy consumption in a scenario where 25% of the IoT applications demand real-time and low-latency services is presented. It is shown that the mean energy expenditure in fog computing is 40.48% less than that in a conventional cloud computing model. Evaluation results show that fog computing is an improved, eco-friendly computing platform that can support IoT better compared to the existing cloud computing paradigm [64]. To take advantage of edge computing and to complement centralized cloud computing, a portion of IoT applications that are energy-efficient in fog computing architecture should be identified. In order to compare the energy consumption of applications using centralized DCs in cloud computing with applications using nano Data Centers (nDCs) based on fog computing, flow-based and time-based energy consumption models for shared and unshared network equipment are proposed in Ref. [65]. Correspondingly, a set of measurements and experiments are used to provide data for the models, in which nano-servers in fog computing are implemented.

The results indicate that the best energy savings using nDCs can be attained for some applications that generate and distribute a large amount of data in end-user premises with low access data rate, such as video surveillance in end-users' homes. The tradeoff between power consumption and transmission delay in the fog-cloud computing system is investigated in Ref. [66]. The segmentation of what tasks go to the edge and what tasks go to the backend cloud is application specific. Simulation results on the user case of medical emergency service demonstrate the benefits of coordinated control and management of a combined fog and cloud system. Thus, the design of a coordinated management strategy becomes critical and needs to address the different cloud/edge resources in a joint framework capable of managing the emerging edge-to-cloud computing and network architecture [67].

4.4. *Fog-RAN: fog computing in radio access networks*

The 5G RAN seamlessly and ubiquitously connects everything, brings a 1000-fold increase in terms of area capacity, supports 100 billion connected wireless devices, and provides diversified use cases as well as high QoS requirements of multimedia applications, compared with current 4G LTE networks [68]. To achieve the above goals, C-RAN has been proposed as a combination of emerging technologies from both the wireless network and cloud computing [69]. The conventional BS is decoupled into two parts: the baseband resources are pooled at BaseBand Units (BBUs) in a centralized location, remote radio heads with radio frequency functions connect with the BBU pool through the wire/-wireless fronthaul links. Based on centralized cloud principle of sharing storage and computing resources via virtualization, Cloud RANs (C-RANs) bring the advantages of high spectral efficiency and energy efficiency while at the same time reducing the cost of network deployment and operation. However, the practical fronthaul is often capacity constrained or time-delay constrained, which presents a bottleneck to the capacity of C-RANs [70]. To overcome the disadvantages of C-RANs with fronthaul constraints, heterogeneous C-RANs (H-CRANs) are proposed as a potential solution [71]. Compared to the C-RAN architecture, the proposed H-CRAN alleviates the fronthaul requirements with the participation of High-Power Nodes (HPNs) (e.g., macro or micro base station). The user and control planes are decoupled in H-CRANs. HPNs execute the functions of the control plane, which delivers all control signaling and system broadcasting data to UEs. HPNs are connected to the BBU pool via the backhaul links, which alleviates the constraints on fronthaul. However, H-CRANs still have the same challenges in practice. More and more IoT applications generate unprecedented amount of data at the edge of networks, which worsens the fronthaul constraints. In addition, processing and storage capabilities in edge devices have not been fully exploited; this is a promising approach to successfully alleviate the burden of the fronthaul and BBU pool. Taking full advantage of fog computing and C-RANs, Fog Radio Access Networks (F-RANs) have been proposed to tackle these aforementioned disadvantages of C-RANs and H-CRANs [72]. In F-RANs, local radio signal processing, cooperative radio resource management, and distributed storage capabilities in edge devices can decrease the heavy burden on the fronthaul and avoid large-scale radio signal processing in the centralized BBU pool. Therefore, F-RANs can achieve high SE/EE, low latency, and excellent reliability to meet 5G requirements.

Table 2 presents the differences between MEC and F-RANs. First, F-RANs have been proposed as an enhancement and evolution of C-RANs to overcome the disadvantages of C-RANs with the fronthaul constraints. MEC is based on a virtualized platform to enable an open RAN, which can host third party innovative applications and content at the edge of the network. Operators can open their networks to authorized third parties, exposing capabilities to over the top players and application developers to flexibly and rapidly deploy innovative applications and services toward mobile subscribers, enterprises, and vertical segments. MEC does not contradict with C-RANs but rather complement them. MEC is mainly compute-oriented, which enables running computation-intensive tasks for edge users by offloading energy-consuming computations of the applications to the MEC server to enhance user experience. In F-RANs, the Cooperative Radio Signal Processing (CRSP) and Cooperative Radio Resource Management (CRRM) can be executed in Fog-computing-based Access Point (F-AP), and the edge caching in F-APs is a key technology to improve SE and EE in maintaining a low latency level [73]. In addition, the MEC server is compatible with the conventional wireless network architecture, which can be deployed at multiple locations, such as at the eNodeB, RNC, and RAT cell aggregation sites. The system architecture of F-RANs is evolved from HetNets and C-RANs, and F-AP is introduced to integrate not only the front Radio Frequency (RF), but also the limited caching, the locally distributed CRSP, and simple CRRM functional capabilities.

Table 2
Differences between MEC and F-RANs.

	MEC	F-RANs
Motivation	Enable an open radio access network that can host third party innovative applications and content at the edge of the network	Overcome the disadvantages of the fronthaul constraints with limited capacity and long delay
Relationship with C-RANs	Complement with C-RANs	An enhancement and evolution of C-RANs
Key technology	Computation offloading	Edge caching
Deployment scheme	Be compatible with conventional wireless network architecture	A new system architecture is evolved from HetNets and C-RANs by introducing F-AP

5. Open issues and challenges

Table 3 summarizes the main features of these three edge computing technologies. Fog computing is initiated to address some challenges in meeting new requirements of IoTs. MEC is recognized as one of the key technologies to meet 5G requirements. Cloudlets are proposed to address some challenges in mobile computing. From the application point of view, MEC enables an open RAN, which can host third party innovative applications and content at the edge of the network. Cloudlet enables new classes of mobile applications that are both compute-intensive and latency-sensitive in an open ecosystem based on cloudlets. Fog computing enables high-performance, interoperability, and security in a multi-vendor fog computing-based ecosystem. The similarity between the three technologies is openness. Operators open their networks to third parties to deploy innovative applications and services.

In this section, some of the pertinent open issues that require additional investigations for edge computing are summarized.

5.1. Big data mining in edge computing

To adapt the massive types of packet traffic and the time-varied radio channel, edge computing should be information-aware and reconfigured. Big data, described by volume, variety, velocity, and value, include subscriber-level, cell-level, core-network-level, and other level data, can facilitate the network toward a more proactive one [74]. Owing to the fast development of big data mining, it is feasible to utilize big data technology to extract interesting patterns or knowledge to enhance the self-organizing capabilities in edge computing. Hierarchical data mining techniques should be used. The transmission of large volume of data collected by edge devices puts a heavy burden on the fronthaul/backhaul, and as a result, data mining can be pre-executed in the edge devices. In the centralized cloud computing, this pre-executed information is reprocessed. Meanwhile, the computing of sparse, uncertain and incomplete data is a big problem, which requires advanced data mining algorithms [75].

5.2. Network slicing in edge computing

To meet the diverse use cases and business models for the emerging applications of mobile internet and IoT, both revolutionary wireless network architectures and advanced technologies are anticipated. As a result, network slicing is proposed recently to flexibly provide SDN in a cost-efficient way. In the concept of network slicing [76], the network entity is sliced into multiple isolated network slice instances, and each slice instance has appropriate network functions and uses advanced radio access technologies for a specific use case or business model. By exploring SDN and NFV, network slice instances and the isolation between them can be conveniently realized [77]. Despite the evident attractive advantages in centralized cloud computing, network slicing comes with several severe challenges when applied in edge computing. First, the conventional creation of network slice instance is mainly business driven. The network slicing solution mainly addresses the requirements of different services, which do not highlight the characteristics of edge computing on network slicing creation. For example, when the radio resource in RANs is in shortage, the requested network slicing may not be effective. As a result, the network slicing should consider radio

Table 3
Comparison of cloudlets, MEC, and fog computing.

Item	Mobile edge computing	Cloudlets	Fog computing
Organization	ETSI MEC supported by Huawei, IBM, Intel, Nokia Networks, NTT DoCoMo, and Vodafone	OEC launched by Vodafone, Intel, Huawei, and Carnegie Mellon University	OpenFog Consortium founded by ARM, Cisco, Dell, Intel, Microsoft, and Princeton University
Is the corresponding consortium a standards body?	Yes	No (influences other standards organizations such as ETSI MEC)	No (OpenFog has an affiliation agreement with IEEE and will be establishing liaisons with other standards organizations, with the objective of collaborating in the creation of standards)
Which business interests are driving?	5G requirements in the telecommunications industry	Some applications based on mobile computing	Internet of Things
Motivation from the application point of view	enable an open RAN that can host third party innovative applications and content at the edge of the network	enable new classes of mobile applications that are both compute-intensive and latency-sensitive in an open ecosystem based on cloudlets	Enable high-performance, interoperability and security in a multi-vendor fog computing-based ecosystem
Features on openness	Operators open their networks to third parties to deploy innovative applications and services	OPENSTACK++ is an open source platform that extends the functionality of Openstack to support cloudlets	The OpenFog Reference Architecture is used as a common baseline for achieving a multivendor interoperable fog computing ecosystem

transmission impacts, and the corresponding network slicing jointly considering the status of RANs should be defined. Second, most of the existing work on network slicing is purely based on CNs, while network slicing as an end-to-end solution should cover the specific characteristics of RANs. To overcome these challenges, a framework solution for a new network slicing technique on edge computing is anticipated.

6. Conclusion

This paper outlines and surveys the state-of-the-art edge computing technologies. With the goal of understanding further intricacies of the key technologies, we have broadly divided the body of knowledge into cloudlet, mobile edge computing, and fog computing. Within each of these aspects, we have given a detailed tutorial on the principle, system architecture, standards, and applications. Nevertheless, given the relative infancy of the field, there are still many outstanding problems that require further investigation from the perspective of key techniques and advanced solutions. Given the extensiveness of the research areas, it is also concluded that more rigorous investigations are required with greater attention to be focused on transforming well-established fog computing into fog computing-based RANs. Furthermore, with the introduction of the advanced big data mining and network slicing, the availability of varied degrees of freedom along with the associated constraints further beckon the design and validation of the original models in the context of edge computing.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61361166005, the National High Technology Research and Development Program of China under Grant No. 2014AA01A701, the National Basic Research Program of China (973 Program) under Grant No. 2013CB336600, and the State Major Science and Technology Special Projects (Grant No. 2016ZX03001020-006).

References

[1] M. Peng, K. Zhang, Recent advances in fog radio access networks: performance analysis and radio resource allocation, IEEE Access J. 4 (2016) 5003–5009.
[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, Internet of things: a survey on enabling technologies, protocols, and applications, IEEE Commun. Surv. Tutor. 17 (2015) 2347–2376.
[3] CISCO, The Internet of Things How the Next Evolution of the Internet Is Changing Everything, White Pap, 2011, http://www.cisco.com/c/dam/en us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf.
[4] M. Chiang, T. Zhang, Fog and IoT: an overview of research opportunities, IEEE Internet Things J. 3 (2016) 854–864.
[5] F. Ganz, D. Puschmann, P. Barnaghi, F. Carrez, A practical evaluation of information processing and abstraction techniques for the internet of things, IEEE Internet Things J. 2 (2015) 340–354.
[6] M.A. Razzaque, M. Milojevic-Jevric, A. Palade, S. Cla, Middleware for internet of things: a survey, IEEE Internet Things J. 3 (2016) 70–95.
[7] A. Brogi, S. Forti, QoS-aware deployment of IoT applications through the fog, IEEE Internet Things J. (2017), https://doi.org/10.1109/JIOT.2017.2701408.
[8] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, W. Zhao, A survey on internet of things: architecture, enabling Technologies, security and privacy, and applications, IEEE Internet Things J. (2017).
[9] M. Satyanarayanan, The emergence of edge computing, Computer (Long. Beach. Calif) 50 (2017) 30–39.
[10] N. Bizanis, F. Kuipers, SDN and virtualization solutions for the Internet of Things: a survey, IEEE Access. J. 4 (2016) 5591–5606.
[11] J. Li, M. Peng, A. Cheng, Y. Yu, C. Wang, Resource allocation optimization for delay-sensitive traffic in fronthaul constrained cloud radio access networks, IEEE Syst. J. (2014) 1–12.
[12] ETSI, Mobile-edge Computing Introductory Technical White Paper, White Paper, Mobile-edge Computing Industry Initiative, 2014, https://portal.etsi.org/portals/0/tbpages/mec/docs/mobile-edge_computing_– _introductory_technical_white_paper_v1.
[13] Y. Mao, C. You, J. Zhang, K. Huang, K.B. Letaief, A Survey on Mobile Edge Computing: the Communication Perspective, 2017. https://arxiv.org/abs/1701.01090.
[14] Guenter I. Klas, Fog Computing and Mobile Edge Cloud Gain Momentum Open Fog Consortium, ETSI MEC and Cloudlets, 2015. http://yucianga.info/?p=938.
[15] M. Satyanarayanan, P. Bahl, R. Caceres, N. Davies, The case for VM-base cloudlets in mobile computing, Pervasive Comput. 8 (2009) 14–23.
[16] Nokia Solutions and Networks, Increasing Mobile Operators' Value Proposition with Edge Computing, White Pap, 2013, pp. 1–6, http://nsn.com/portfolio/liquid-net/intelligent-broadband-management/liquid-applications.
[17] Y. Ch. Hu, M. Patel, D. Sabella, N. Sprecher, V. Young, Mobile Edge Computing a Key Technology towards 5G, 2015. http://10.3.200.202/cache/8/03/etsi.org/6e14a9668574b8b935111768d9f6e501/etsi_wp11_mec_a_key_technology_towards_5g.pdf.
[18] N. Sprecher, J. Friis, R. Dolby, J. Reister, Edge Computing Prepares for a Multi-access Future, 2016. http://www.telecomtv.com/articles/mec/edge-computing-prepares-for-a-multi-access-future-13986/.
[19] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog Computing and its Role in the Internet of Things, Proc. First Ed. MCC Work. Mob. Cloud Comput., 2012, pp. 13–16.
[20] Z. Sanaei, S. Abolfazli, A. Gani, R. Buyya, Heterogeneity in mobile cloud computing: taxonomy and open challenges, IEEE Commun. Surv. Tutor. 16 (2014) 369–392.
[21] K. Ha, P. Pillai, W. Richter, Y. Abe, M. Satyanarayanan, Just-in-time provisioning for cyber foraging, in: MobiSys 2013-Proc. 11th Annu. Int. Conf. Mob. Syst. Appl. Serv, 2013, pp. 153–166.
[22] Y. Gao, W. Hu, K. Ha, B. Amos, et al., Are Cloudlets Necessary? School of Computer Sci. Carnegie Mellon Univ., Pittsburgh, PA, USA, 2015. Tech. Rep. CMU-CS-15-139, http://reports-archive.adm.cs.cmu.edu/anon/anon/usr/ftp/usr0/ftp/2015/CMU-CS-15-139.pdf.
[23] K. Ha, Y. Abe, Z. Chen, W. Hu, et al., Adaptive VM Handoff across Cloudlets, Technical Report CMU-CS-15-113, CMU School of Computer Science, 2015, http://ra.adm.cs.cmu.edu/anon/2015/CMU-CS-15-113.pdf.
[24] M. Satyanarayanan, R. Schuster, M. Ebling, G. Fettweis, H. Flinck, K. Joshi, K. Sabnani, An open ecosystem for mobile-cloud convergence, IEEE Commun. Mag. 53 (2015) 63–70.
[25] K. Ha, Z. Chen, W. Hu, W. Richter, P. Pillai, M. Satyanarayanan, Towards wearable cognitive assistance, in: Proc. 12th Annu. Int. Conf. Mob. Syst. Appl. Serv. - MobiSys'14, 2014, pp. 68–81.
[26] M. Satyanarayanan, P. Simoens, Y. Xiao, P. Pillai, Z. Chen, K. Ha, W. Hu, B. Amos, Edge analytics in the internet of things, IEEE Pervasive Comput. 14 (2015) 24–31.
[27] M. Satyanarayanan, G. Lewis, E. Morris, S. Simanta, J. Boleng, K. Ha, The role of cloudlets in hostile environments, IEEE Pervasive Comput. 12 (2013) 40–49.

[28] K. Ha, M. Satyanarayanan, OpenStack++ for Cloudlet Deployment, School of Computer Science Carnegie Mellon University Pittsburgh, 2015. http://elijah.cs.cmu.edu/DOCS/CMU-CS-15-123.pdf.
[29] ETSI GS MEC 001, Mobile Edge Computing (MEC) Terminology V1.1.1, 2016. http://www.etsi.org/deliver/etsi_gs/MEC/001_099/001/01.01.01_60/gs_MEC001v010101p.pdf.
[30] ETSI GS MEC 002, Mobile Edge Computing (MEC) Technical Requirements V1.1.1, 2016. http://www.etsi.org/deliver/etsi_gs/MEC/001_099/002/01.01.01_60/gs_MEC002v010101p.pdf.
[31] ETSI GS MEC 003, Mobile Edge Computing (MEC) Framework and Reference Architecture V1.1.1, 2016. http://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf.
[32] ETSI GS MEC-IEG 004, Mobile Edge Computing (MEC) Service Scenarios V1.1.1, 2015. http://www.etsi.org/deliver/etsi_gs/MEC/001_099/004/01.01.01_60/gs_MEC003v010101p.pdf.
[33] ETSI GS MEC-IEG 005, Mobile Edge Computing (MEC) Proof of Concept Framework V1.1.1, 2015. http://www.etsi.org/deliver/etsi_gs/MEC-IEG/001_099/005/01.01.01_60/gs_MEC-IEG005v010101p.pdf.
[34] ETSI GS MEC-IEG 006, Mobile Edge Computing Market Acceleration MEC Metrics Best Practice and Guidelines V1.1.1, 2017. http://www.etsi.org/deliver/etsi_gs/MEC-IEG/001_099/006/01.01.01_60/gs_MEC-IEG006v010101p.pdf.
[35] T. Taleb, S. Dutta, A. Ksentini, M. Iqbal, H. Flinck, Mobile edge computing potential in making cities smarter, IEEE Commun. Mag. 55 (2017) 38–43.
[36] S. Lin, H.F. Cheng, W. Li, Z. Huang, P. Hui, C. Peylo, Ubii: physical world interaction through augmented reality, IEEE Trans. Mob. Comput. 16 (2017) 872–885.
[37] N. Kumar, S. Zeadally, J.J.P.C. Rodrigues, Vehicular delay-tolerant networks for smart grid data management using mobile edge computing, IEEE Commun. Mag. 54 (2016) 60–66.
[38] X. Sun, N. Ansari, EdgeIoT: mobile edge computing for the Internet of Things, IEEE Commun. Mag. 54 (2016) 22–29.
[39] A. Ceselli, M. Premoli, S. Secci, Mobile edge cloud network design optimization, IEEE/ACM Trans. Netw. 25 (2017) 1818–1831.
[40] S. Barbarossa, S. Sardellitti, P. Di Lorenzo, Communicating while computing: distributed mobile cloud computing over 5G heterogeneous networks, IEEE Signal Process. Mag. 31 (2014) 45–55.
[41] P. Mach, Z. Becvar, Mobile Edge Computing: a survey on architecture and computation offloading, IEEE Commun. Surv. Tutor. 19 (2017) 1628–1656, https://doi.org/10.1109/COMST.2017.2682318.
[42] H. Flores, P. Hui, S. Tarkoma, Y. Li, S. Srirama, R. Buyya, Mobile code offloading: from concept to practice and beyond, IEEE Commun. Mag. 53 (2015) 80–88.
[43] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, Y. Zhang, Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks, IEEE Access. J. 4 (2016) 5896–5907.
[44] X. Chen, L. Jiao, W. Li, X. Fu, Efficient multi-user computation offloading for mobile-edge cloud computing, IEEE/ACM Trans. Netw. 24 (2016) 2795–2808.
[45] C. You, K. Huang, H. Chae, B.H. Kim, Energy-efficient resource allocation for mobile-edge computation offloading, IEEE Trans. Wirel. Commun. 16 (2017) 1397–1411.
[46] S. Sardellitti, G. Scutari, S. Barbarossa, Joint optimization of radio and computational resources for multicell mobile-edge computing, IEEE Trans. Signal Inf. Process. Over Netw. 1 (2015) 89–103.
[47] S. Secci, P. Raad, P. Gallard, Linking virtual machine mobility to user mobility, IEEE Trans. Netw. Serv. Manag. 13 (2016) 927–940.
[48] P. Raad, S. Secci, D.C. Phung, A. Cianfrani, P. Gallard, G. Pujolle, Achieving sub-second downtimes in large-scale virtual machine migrations with LISP, IEEE Trans. Netw. Serv. Manag. 11 (2014) 133–143.
[49] T. Taleb, A. Ksentini, P. Frangoudis, Follow-me cloud: when cloud services follow mobile users, IEEE Trans. Cloud Comput. (2016), 1–1.
[50] X. Sun, N. Ansari, PRIMAL: PRofIt Maximization Avatar pLacement for mobile edge computing, in: 2016 IEEE Int. Conf. Commun. ICC 2016, 2016, pp. 1–6.
[51] J. Plachy, Z. Becvar, E.C. Strinati, Dynamic resource allocation exploiting mobility prediction in mobile edge computing, in: IEEE Int. Symp. Pers. Indoor Mob. Radio Commun. PIMRC, 2016.
[52] Open Fog Consortium, [Online]. https://www.openfogconsortium.org/.
[53] OpenFog Consortium Architecture Working Group, OpenFog Architecture Overview White Paper. https://www.openfogconsortium.org/wp-content/uploads/OpenFog-Architecture-Overview-WP-2-2016.pdf.
[54] Open Fog Consortium, OpenFog Reference Architecture for Fog Computing. https://www.openfogconsortium.org/wp-content/uploads/OpenFog_Reference_Architecture_2_09_17-FINAL-1.pdf.
[55] A.V. Dastjerdi, R. Buyya, Fog computing: helping the Internet of Things realize its potential, Computer 49 (2016) 112–116.
[56] S.K. Sharma, X. Wang, Live data analytics with collaborative edge and cloud processing in wireless IoT networks, IEEE Access 5 (2017) 4621–4635.
[57] B. Tang, Z. Chen, G. Hefferman, S. Pei, W. Tao, H. He, Q. Yang, Incorporating intelligence in fog computing for big data analysis in smart cities, IEEE Trans. Ind. Inf. 13 (2017) 2140–2150, https://doi.org/10.1109/TII.2017.2679740.
[58] S. Basudan, X. Lin, K. Sankaranarayanan, A privacy-preserving vehicular crowdsensing based road surface condition monitoring system using fog computing, IEEE Internet Things J. 4 (2017) 772–782.
[59] P. Hu, H. Ning, T. Qiu, H. Song, Y. Wang, X. Yao, Security and privacy preservation scheme of face identification and resolution framework using fog computing in internet of things, IEEE Internet Things J. (2017), 1–1.
[60] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, S. Chen, Vehicular fog computing: a viewpoint of vehicles as the infrastructures, IEEE Trans. Veh. Technol. 65 (2016) 3860–3873.
[61] M. Aazam, E.N. Huh, Fog computing: the cloud-IoT/IoE middleware paradigm, IEEE Potentials 35 (2016) 40–44.
[62] L. Gu, D. Zeng, S. Guo, A. Barnawi, Y. Xiang, Cost efficient resource management in fog computing supported medical cyber-physical system, IEEE Trans. Emerg. Top. Comput. 5 (2017) 108–119.
[63] S. Misra, S. Sarkar, Theoretical modelling of fog computing: a green computing paradigm to support IoT applications, IET Netw. 5 (2016) 23–29.
[64] S. Sarkar, S. Chatterjee, S. Misra, Assessment of the suitability of fog computing in the context of internet of things, IEEE Trans. Cloud Comput. (2015).
[65] F. Jalali, K. Hinton, R. Ayre, T. Alpcan, R.S. Tucker, Fog computing may help to save energy in cloud computing, IEEE J. Sel. Areas Commun. 34 (2016) 1728–1739.
[66] R. Deng, R. Lu, C. Lai, T.H. Luan, H. Liang, Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption, IEEE Internet Things J. (2016) 1171–1181.
[67] X. Masip-Bruin, E. Marín-Tordera, G. Tashakor, A. Jukan, G.J. Ren, Foggy clouds and cloudy fogs: a real need for coordinated management of fog-to-cloud computing systems, IEEE Wirel. Commun. 23 (2016) 120–128.
[68] M. Peng, Y. Li, Z. Zhao, C. Wang, System architecture and key technologies for 5G heterogeneous cloud radio access networks, IEEE Netw. 29 (2015) 6–14.
[69] M. Peng, Y. Sun, X. Li, Z. Mao, C. Wang, Recent advances in cloud radio access networks: system architectures, key techniques, and open issues, IEEE Commun. Surv. Tutor. 18 (2016) 2282–2308.
[70] M. Peng, C. Wang, V. Lau, H.V. Poor, Fronthaul-constrained cloud radio access networks: insights and challenges, IEEE Wirel. Commun. 22 (2015) 152–160.
[71] M. Peng, Y. Li, J. Jiang, J. Li, C. Wang, Heterogeneous cloud radio access networks: a new perspective for enhancing spectral and energy efficiencies, IEEE Wirel. Commun. 21 (2014) 126–135.
[72] M. Peng, S. Yan, K. Zhang, C. Wang, Fog-computing-based radio access networks: issues and challenges, IEEE Netw. 30 (2016) 46–53.
[73] S. Jia, Y. Ai, Z. Zhao, M. Peng, C. Hu, Hierarchical content caching in fog radio access networks: ergodic rate and transmit latency, China Commun. 13 (2016) 1–14.
[74] A. Imran, A. Zoha, Challenges in 5G: how to empower SON with big data for enabling 5G, IEEE Netw. 28 (2014) 27–33.
[75] X. Wu, X. Zhu, G.Q. Wu, W. Ding, Data mining with big data, IEEE Trans. Knowl. Data Eng. 26 (2014) 97–107.
[76] R. Hattachi, J. Erfanian, 5G White Paper, NGMN Alliance, 2015.
[77] Q. Li, G. Wu, A. Papathanassiou, U. Mukherjee, An End-to-end Network Slicing Framework for 5G Wireless Communication Systems, 2016. https://arxiv.org/abs/1608.00572.